AFRL-VS-TR-2002-1675

INSTALLATION AND OPERATION OF PARTICLE TRANSPORT SIMULATION PROGRAMS TO MODEL THE DETECTION AND MEASUREMENT OF SPACE RADIATION BY SPACE-BORNE SENSORS

Stanley Woolf

ARCON Corporation
260 Bear Hill Rd.
Waltham, MA 02451-1080

3 January 2002

Scientific Report No. 2

20030915 111

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

AIR FORCE RESEARCH LABORATORY
Space Vehicles Directorate
29 Randolph Rd
AIR FORCE MATERIEL COMMAND
Hanscom AFB, MA 01731-3010

This technical report has been reviewed and is approved for publication.

Contract Manager

Branch Chief

This document has been reviewed by the ESC Public Affairs Office and has been approved for release to the National Technical Information Service.

Qualified requestors may obtain additional copies form the Defense Technical Information Center (DTIC). All others should apply to the National Technical Information Service.

If your address has changed, if you with to be removed from the mailing list, or if the addressee is no longer employed by your organization, please notify AFRL/VSIM, 29 Randolph Rd., Hanscom AFB, MA 01731-3010. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
03-01-2002	Scientific Report No. 2	1 Aug 2000 - 31 July 2001

4. TITLE AND SUBTITLE
Installation and Operation of Particle Transport Simulation Programs to Model the Detection and Measurement of Space Radiation by Space-Borne Sensors

5a. CONTRACT NUMBER F1962899-C-0077

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER 63401F

6. AUTHOR(S)
Dr. Stanley Woolf

5d. PROJECT NUMBER 2822

5e. TASK NUMBER GC

5f. WORK UNIT NUMBER AR

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
ARCON Corporation
260 Bear Hill Road
Waltham, MA 02451-1080

8. PERFORMING ORGANIZATION REPORT NUMBER
UTD-STR-A002-01-630117

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFRL/VSBXR (Bronek Dichter)
29 Randolph Road
Hanscom AFB, MA 01731-3010

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-VS-TR-2002-1675

12. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution Unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This is a report of technical progress made during 1 Aug 00 to 31 Jul 01 in the areas of: (1) research and evaluation of particle transport simulation programs for modeling the detection and measurement of space radiation by space-borne sensors; (2) construction of realistic flight sensor computer models; (3) performance of particle transport calculation; (4) analysis of transport simulation results, including single particle tracking; (5) addition of new capabilities such as single particle tracking and specialized source geometry to an existing particle transport simulation program; (6) space-borne dosimeter simulation studies; (7) three-dimensional visualization of ITS-ACCEPT and MCNPX were applied to the modeling of the geometry files. The computer programs ITS-ACCEPT and MCNPX were applied to the modeling of the CEASE and HEP sensors. Shown in this report are listings of input files with geometry/materials drawings for the various simulation programs, annotated computer code listings showing program modifications and partial listings of computer code outputs for individual particle tracking and coincidence event identifications.

15. SUBJECT TERMS
Electron and proton transport, Monte Carlo simulation, Particle tracking, Space-borne sensor modeling, ITS-ACCEPT code, MCNPX code, Visualization, Dosimeters

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
U.	U	U	U	96	Bronek Dichter
					19b. TELEPHONE NUMBER *(Include area code)* (781) 377-3991

Table of Contents

1. Introduction

The effort to be described in this report was performed as partial fulfillment of two primary objectives: (1) perform computer simulations of charged particle transport, energy and charge deposition in satellite-borne instrumentation used in research efforts of the Air Force Research Laboratory/ Space Weather Center of Excellence (AFRL/VSBXR) to detect and characterize (by type, energy, intensity, *etc.*) particles associated with ionizing radiation in space; (2) transfer this simulation capability to AFRL/VSBXR and provide advice to Air Force researchers on its use; and (3) create and install additional capabilities in existing Monte Carlo transport programs to: (a) simulate a number of source geometries encountered in the VSXBR research program; and (b) permit "observation" of individual electron track histories.

During this reporting period we worked with the Monte Carlo simulation programs listed below at ARCON and provided assistance and guidance for their use at AFRL. The Monte Carlo transport simulations programs that were used at both ARCON and AFRL in this effort are:

- "ITS 3.0 – Integrated TIGER Series of Coupled Electron/Photon Monte Carlo Code System" [1] - ACCEPT – General three-dimensional transport code
- "MCNPX, Version 2.1.5 – Monte Carlo transport code for neutrons, photons, electrons, mesons, protons, deuterons, tritons, ^{3}He, alpha" [2]

In addition to the two codes listed above, we also acquired a three dimensional geometry and visualization program, Sabrina[3], that while written primarily for use with the MCNP code series, can also be used for reading and writing geometry files for ITS/ACCEPT.

In the following sections, we briefly discuss the computer programs listed above, some of their interrelationships, and provide descriptions and examples of our application of these codes to the modeling of particle transport and trajectory tracking in the CEASE[4] and HEP[5] instruments.

2. Electron Transport Modeling

2.1 Electron Energy Deposition Calculations in Silicon Wafers

Transport calculations for 4 MeV and 6 MeV electrons incident on rectangular silicon dosimeter wafers (0.05cm × 0.9cm × 0.9cm). Twelve source geometry configurations were assumed for each source energy. Duplicate simulation runs were made with both ITS-ACCEPT[1] and MCNPX[2]. The result of performing these calculations accomplished two objectives: 1) provide a set of input files for both simulation programs that could be modified, if desired, and used by AFRL personnel for performing these and similar simulations; and 2) compare the relative advantages and disadvantages of the ACCEPT and MCNPX codes for electron transport. The input files were also set up to produce electron pulse-height spectra. We provided interpretation of the pulse-height spectra results for both codes, the presentations of which are formatted differently, and showed for all practical purposes, the equivalence of their answers.

For all source configurations and both source energies, the Monte Carlo runs were made using 200,000 case histories. The source geometries consisted of: normally incident electron

beams (or point sources); point isotropic sources; disk sources-normal incidence; disk sources-isotropic incidence. All sources were located on the wafer surfaces, either on the 0.9cm × 0.9cm surface (Fig.1a) or on the 0.05cm × 0.9cm surface (Fig. 1b). The input and output files for all 24 Monte Carlo runs were provided to AFRL. Default values for the electron low energy cut-off were used with both ACCEPT ($0.05E_{source}$) and MCNPX (1.0 keV). Since the default value for the MCNPX cut-off energy was set much lower than the ACCEPT value, the run times for MCNPX (~1 hr) far exceeded those for ACCEPT (~0.5 min) by two orders of magnitude. When the same electron cut-off energy was used in MCNPX, the run time was found to be a factor of 1.5 greater than that required for ACCEPT. With the current version of ACCEPT, the lowest electron cut-off energy allowed by the code is 24.5 keV. When the program was run with this cut-off, the run time for 200000 histories increased to 1.17 minutes with no significant change in the results. To achieve the low energy cut-off value of 1 keV, it would have been necessary to use ACCEPTP and XGENP, the P-code versions of ACCEPT and XGEN, containing low-energy electron physics. ACCEPT energy deposition results are shown in Figs.2-6 for 6 MeV sources. Fig. 2 displays the total energy deposited in the silicon wafer shown in Fig.1a for the eight source configurations described and labeled in Table 1.

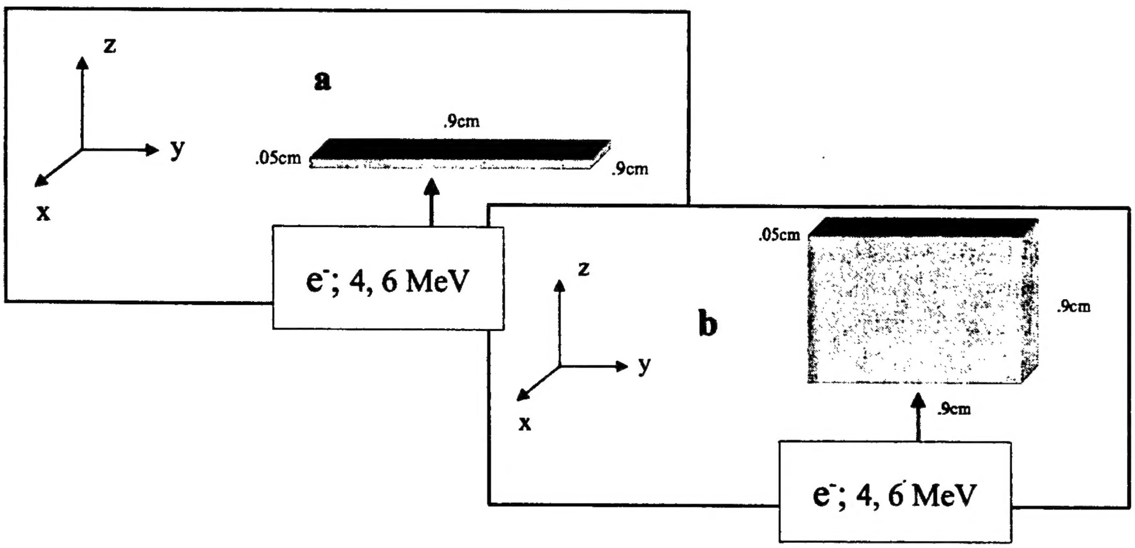
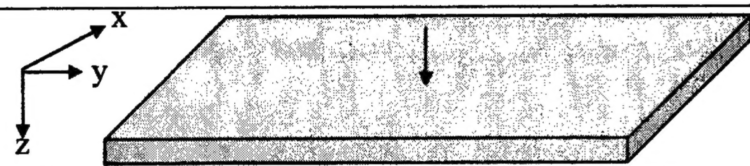
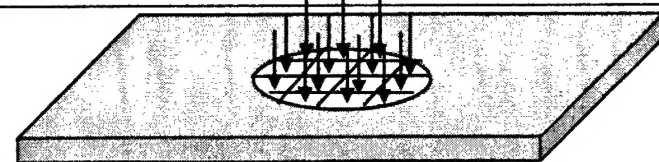
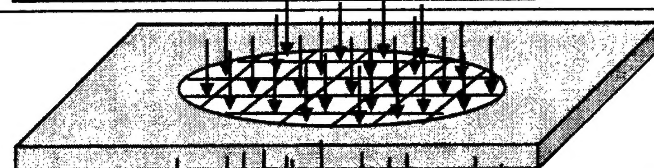
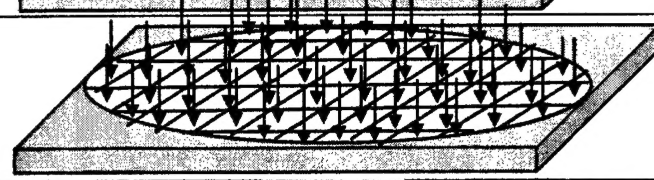
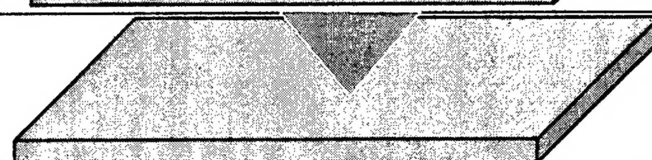
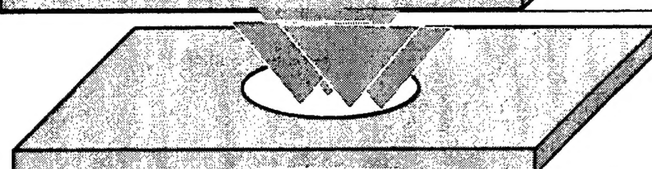
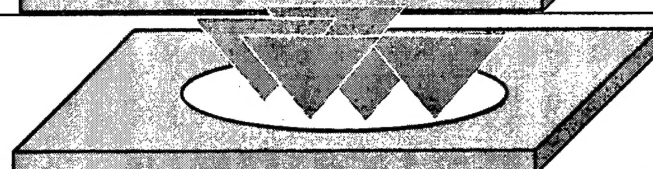
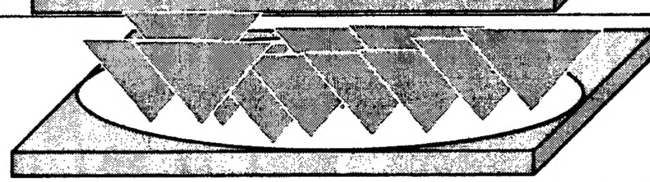


Figure 1. Electron sources incident on silicon dosimeter wafer

Table 1.
Source Configurations for Transport Simulation of 4.0 and 6.0 MeV Electrons in Silicon

Electron Soure Configuration	**Source Geometry**
(1) Single Beam, Normal Incidence at (.45,.45,0.)	
(2) Disk Source, Rad. = 0.1cm, Normal Incidence centered at (.45,.45,0.)	
(3) Disk Source, Radius = 0.2cm, Normal Incidence centered at (.45,.45,0.)	
(4) Disk Source, Radius = 0.449cm, Normal Incidence centered at (.45,.45,0.)	
(5) Point Isotropic, 45° cone centered at (.45,.45,0.)	
(6) Disk Source, Rad. = 0.1cm, Isotropic 45° cones centered at (.45,.45,0.)	
(7) Disk Source, Rad. = 0.2cm, Isotropic 45° cones centered at (.45,.45,0.)	
(8) Disk Source, Rad. = 0.449cm, Isotropic 45° cones centered at (.45,.45,0.)	

The choice of 200000 histories resulted in poor statistics for the point sources, except in the immediate vicinity of the source point. The statistics were much improved, however, everywhere in the silicon wafer (~1%-5% estimated standard error) with the use of spatially

uniform disk sources. The number of histories (200000) was chosen as an expedient to ensure input file correctness. The run files corresponding to the source configurations shown in Table 1 were turned over to AFRL for production runs.

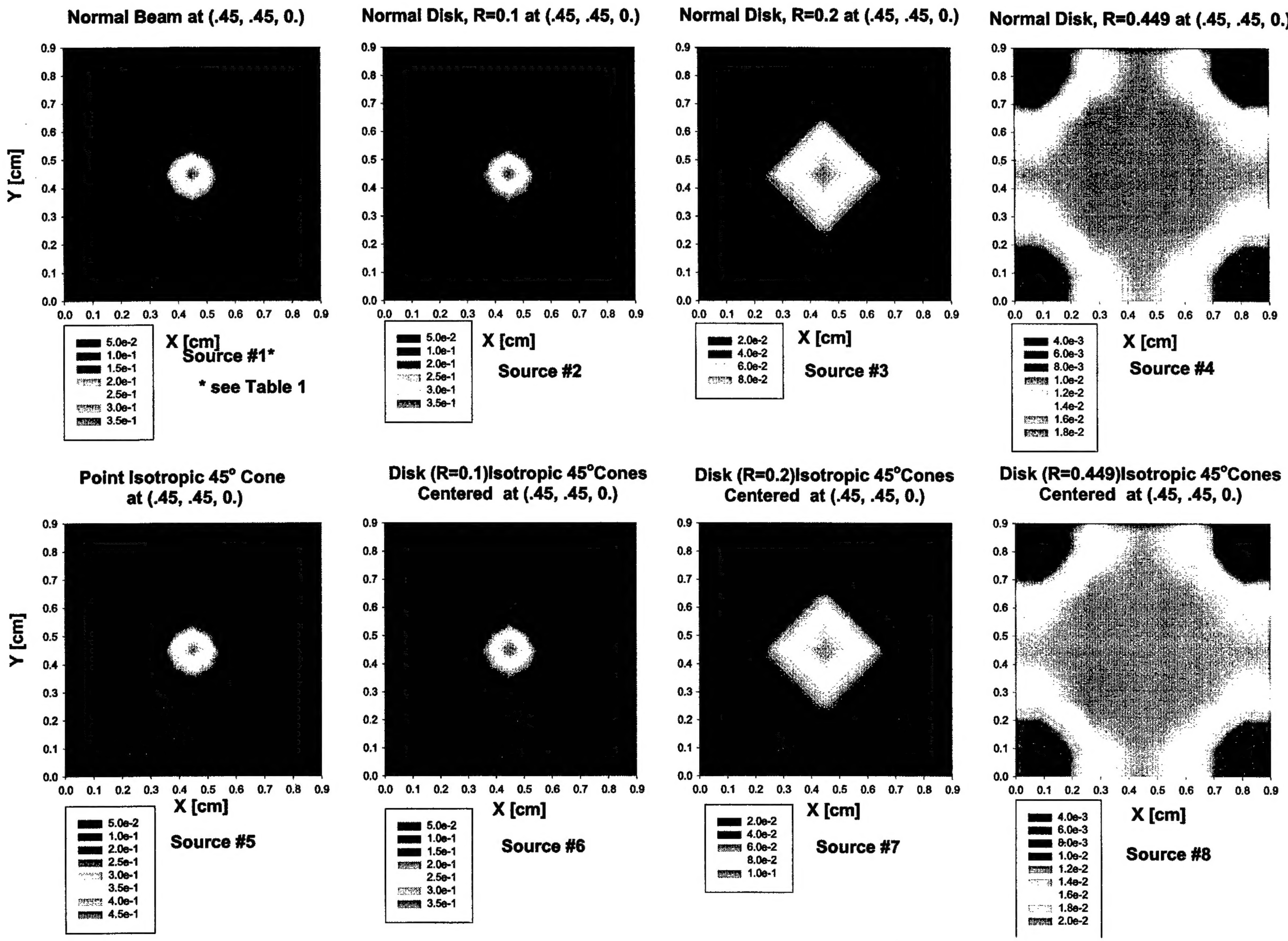
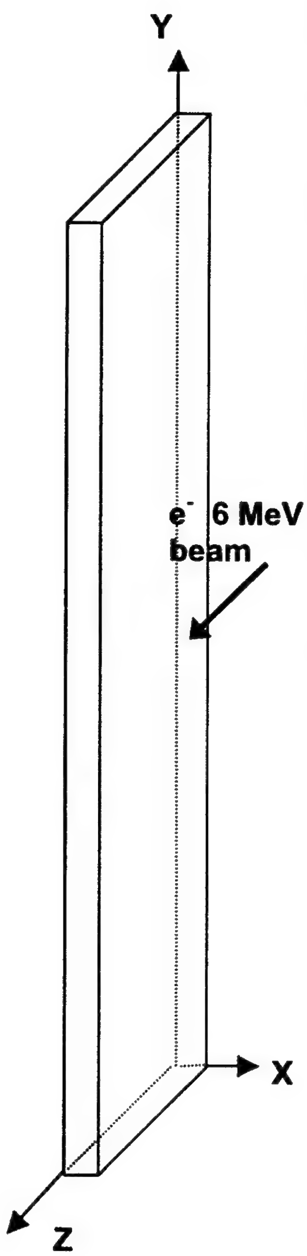


Figure 2. Total energy deposition [MeV] in Si wafer ($0.05\times0.9\times0.9 cm^3$, see Figure 1a) for 8 source geometries; electron source energy = 6 MeV



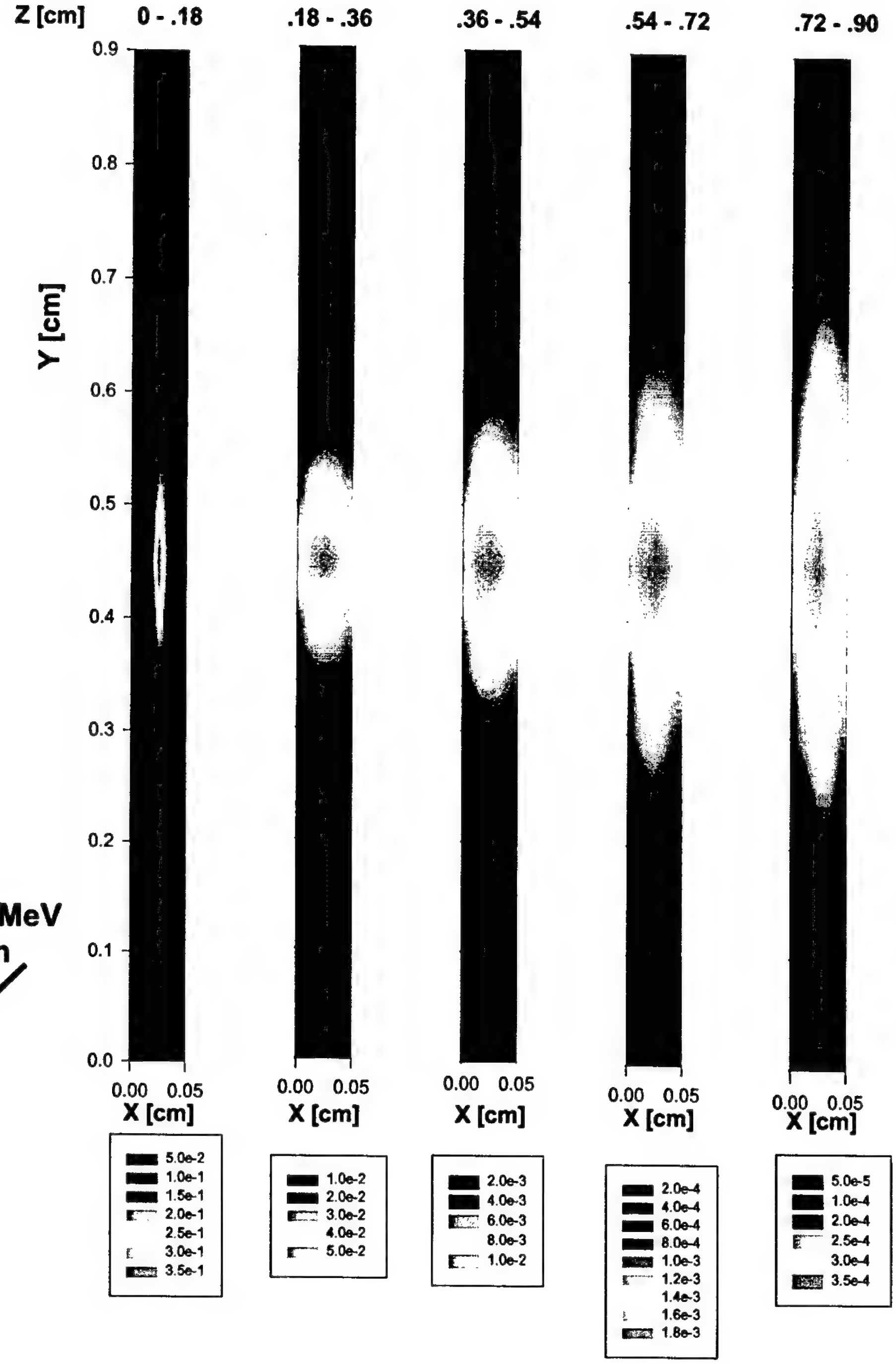
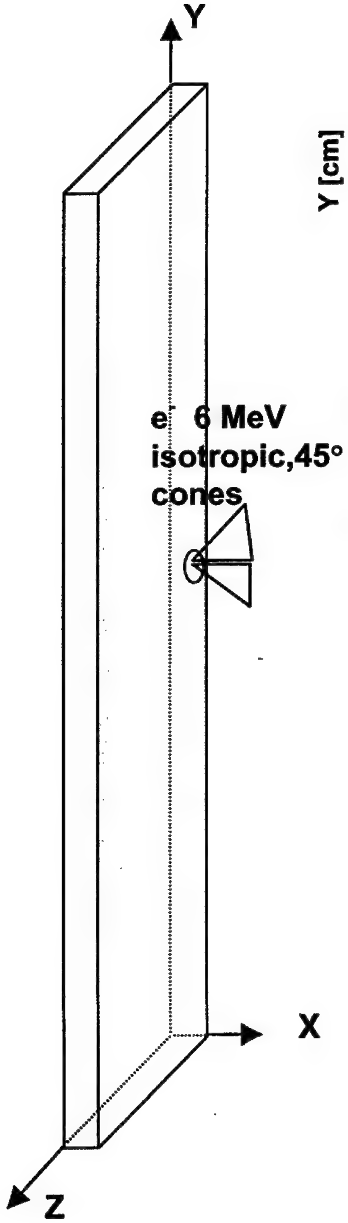


Figure 3. Energy deposition [MeV] in 5 Z-lLayers in Si wafer ($0.05\times0.9\times0.9 cm^3$), 6 MeV electron beam source, normal incidence at (0.025,0.45,0.0); ITS-ACCEPT calculation



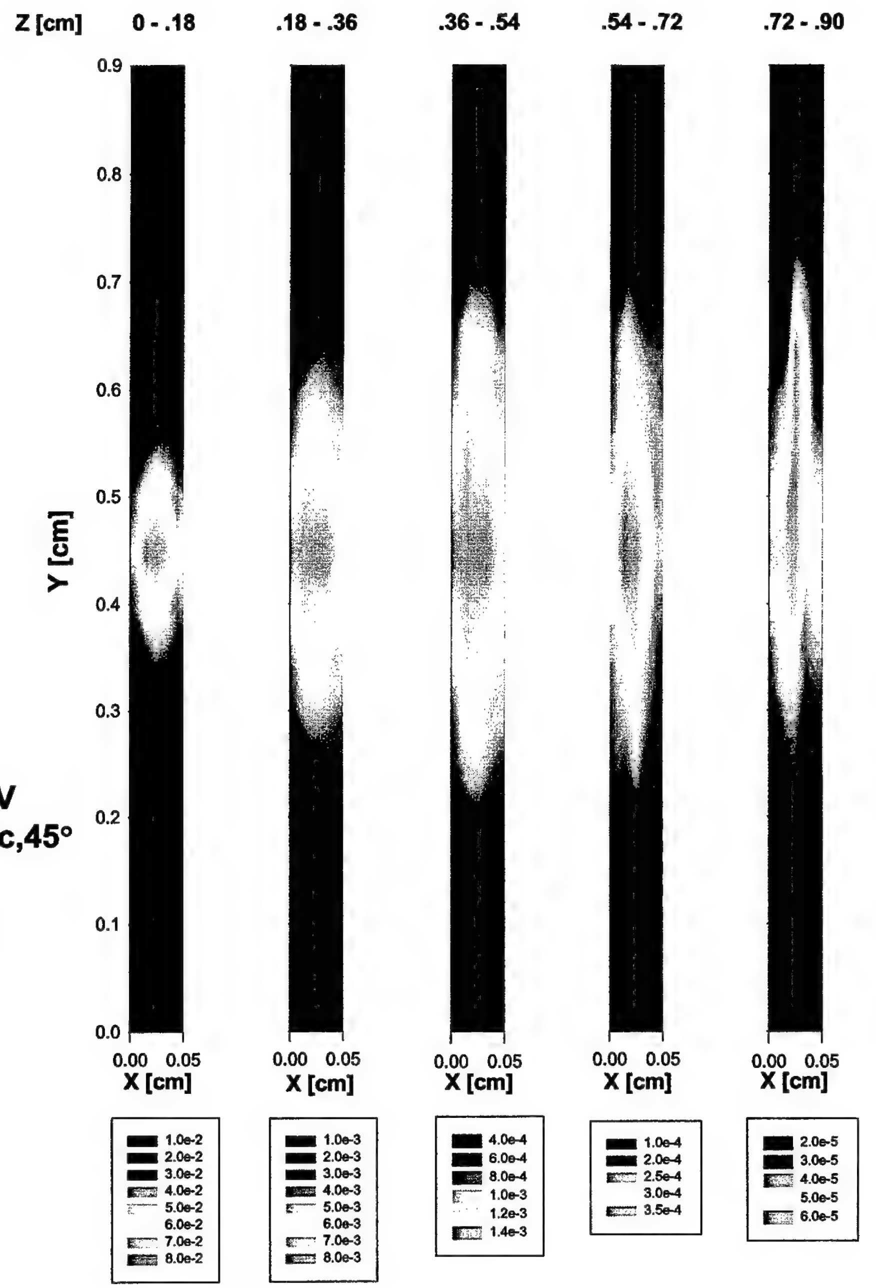


Figure 4. Energy deposition [MeV] in 5 Z-layers in Si wafer ($0.05\times0.9\times0.9cm^3$), 6 MeV electron isotropic 45 deg. cone sources, uniformly distributed on disk (R=0.02cm) centered at (0.025,0.45,0.0); ITS-ACCEPT calculation

2.2 ITS-ACCEPT Electron/Photon Transport Simulations for the HEP Instrument

A new ACCEPT geometry input file for the in-flight version of the HEP instrument was created from a complete set of manufacturing drawings supplied by Amptek, Inc. [5]. This geometry description exactly mimics the geometry description written during the first year of this effort for MCNPX [6]. The ACCEPT geometry description, which contains the same degree of detail as the manufacturing drawings, permits us to: (1) take advantage of the speed and efficiency of the ACCEPT code for performing coupled electron-photon transport calculations in complicated structures; and (2) use the ACCEPT code to confirm the validity of the MCNPX geometry description by comparing electron transport results obtained with the two programs. Unlike ACCEPT, MCNPX can be used to perform transport calculations for protons, neutrons, mesons, and other particle species. It is therefore important to have the ACCEPT version of the HEP geometry to use as an independent verification of the MCNPX geometry description.

Figure 5 shows a cut-away view of the HEP sensor. This illustration was drawn using SABRINA [3] an interactive, three-dimensional geometry visualization and modeling program

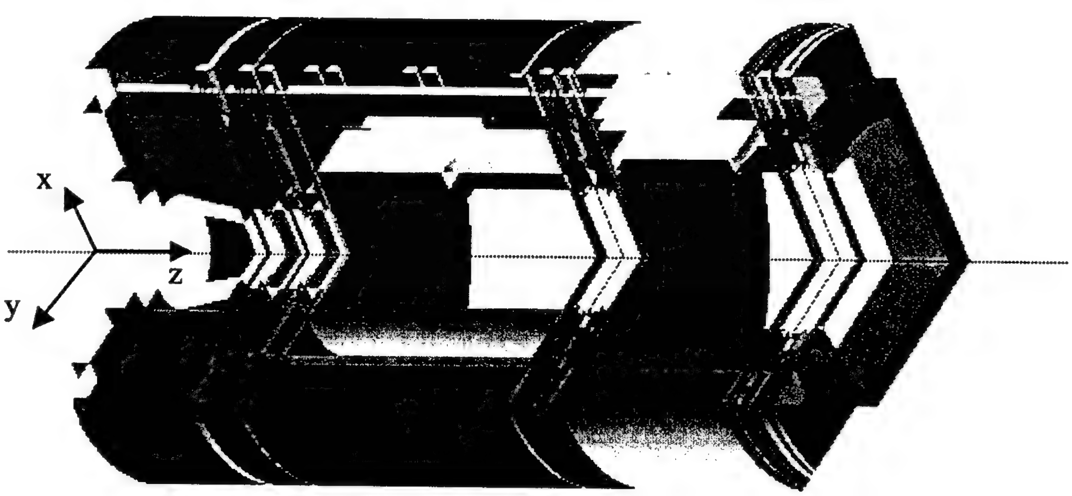


Figure 5. Sabrina [3] rendering of ITS-ACCEPT [1] geometry model of HEP [5] Flight Sensor

that can be used to construct and visualize geometry models for both MCNPX and ACCEPT. Additional useful features of the SABRINA program are: automatic conversion of ACCEPT models to the MCNPX format and particle track ray-tracing.

The ACCEPT input file corresponding to the HEP flight sensor depicted in Figure 5 is listed in Appendix 1. Several ACCEPT runs were made to test the robustness of the geometry file. The purpose was to uncover "holes" (errors) in the geometry specification that are not immediately obvious from the pictures generated with SABRINA. An efficient way to determine which, if any, cells are improperly defined in the input file is to run a large number of case histories for several source configurations and energies. The diagnostic messages that appear in the ACCEPT output when a particle has "lost its way" are of limited value. Because geometry file flaws are usually manifested by abrupt program halts when ten particles have been lost, we have found that the most effective method for constructing highly detailed geometry files is to build up the model in gradual stages of complexity. Test runs of ACCEPT were made using this gradual approach until the model illustrated in Figure 5 passed all tests for robustness. The run file consisting of 318 geometric bodies defining 505 material cells is listed in Appendix 4. This file was used to simulate a 25 MeV electron disk source normally incident on the front face ($z = 0$ plane) of the instrument.

2.3 ITS-ACCEPT Program Enhancements

The ITS-ACCEPT Monte Carlo program modifications that were made fall into two categories: the addition of new source options; and the addition of code that permits the user to view the energy deposition contributions of individual electron tracks. The primary motivation for the source option enhancement was supplied by the fact that the standard disk source option in ACCEPT does not allow for the specification of electron source beam slant angles without slanting the source plane. Source electrons emanating from a plane with off-normal angles of incidence could not originate at equidistant points from the target. The demonstration of this is given in a set of electron transport runs that were made for the aluminum-void-silicon slab geometry shown in Figure 6. Runs were made for six source disk slant angles (θ = 0, 15, 30, 45, 60, 75°) The electron source energy in all cases was 3.5 MeV. Because of the manner in which the standard disk source option in ACCEPT is implemented, it was necessary to adjust the position of the disk center for each θ value in order to ensure that the source disk not intersect with the target medium. An ACCEPT input data file corresponding to the illustration of Figure 6 with $\theta = 45°$ is shown in Figure 7. For the case shown, the radius R of the disk source was set at 1.0 cm. For the 45° slant source it was necessary to place the coordinates of the disk center at $X_s = 2.5$, $Y_s = 2.5$, $Z_s = -0.7071067$ to avoid having source electrons originate inside the target medium. While the ACCEPT code does not permit this source condition, it could be modified to allow sources originating in the interior of a scattering medium.

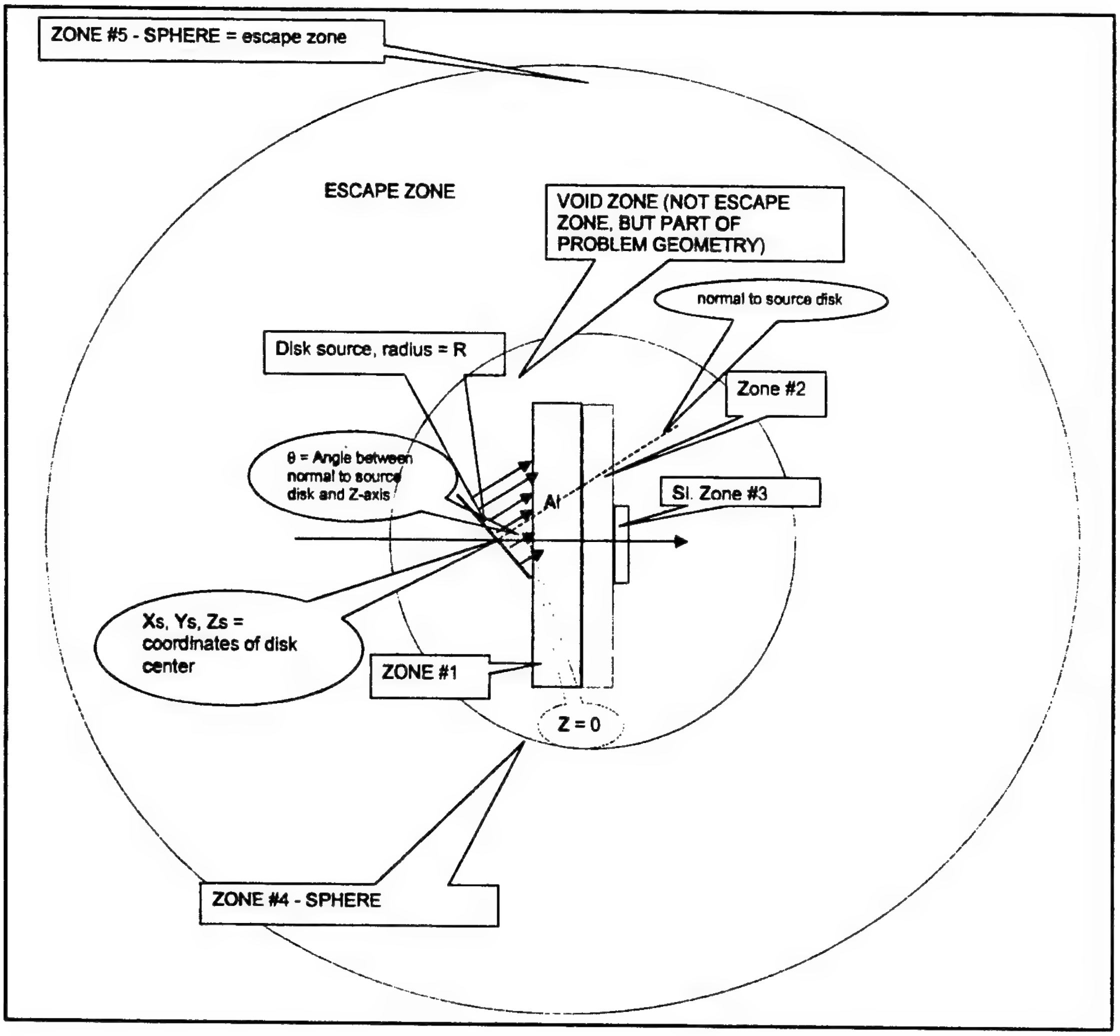


Figure 6. Aluminum / void / silicon ACCEPT problem geometry (not drawn to scale) with slant disk source as described in the input data file shown in Figure 7.

2.3.1 Disk and Rectangle Source Options

A capability was added to the ACCEPT program to allow the use of disk and rectangular spatially uniform distributed sources (electrons or photons) with provision for changing the slant angle of the source beam direction without slanting the source plane itself. These new options are now implemented in the code by the addition of:

(1) a line such as

```
"RECTANGLE-SOURCE   2.05 2.95 2.05 2.95 0.0 0.0"
```

to the ACCEPT input file for the rectangle source case. The 6 numbers (required) are the bounding coordinates of the source rectangle (`XMIN,XMAX,YMIN,YMAX,ZMIN,ZMAX`);

and (2)

```
"CIRCLE-SOURCE   2.5 2.5 0.0 3.5 2.5 0.0"
```

to the ACCEPT input file for the disk source case. The 6 (required) numbers are the `X,Y,Z` coordinates of the disk center and a point on the disk circumference. The program computes the

source disk radius and checks internal consistency by comparing this value with the value entered on the "RADIUS" input line. In both the rectangle and circle source cases, if an error in the input data violates the conditions for geometric validity, informative diagnostics messages are printed in the ACCEPT output file, and the run is aborted. One restriction in the use of these options is that the orientation of the source planes cannot be arbitrarily chosen. Their orientation must be perpendicular to any one of the three Cartesian coordinate axes. Since the choice of orientation of the source beam is allowed to be arbitrary, this restriction, which greatly simplified the re-programming of ACCEPT, does not result in sacrifice of utility.

Test runs were made for both the rectangle and disk source cases. The disk source results were validated by matching the old disk source results for normal incidence.

```
TITLE
  3.5 MEV ON AL/VOID/SI,SLANT DISK SOURCE(THETA=45deg)ZS=-.7071
*********************** GEOMETRY ***********************
GEOMETRY
*1
  RPP    0.000  5.000  0.000  5.000  0.000  0.635
*2
  RPP    0.000  5.000  0.000  5.000  0.635  0.792
*3
  RPP    2.050  2.950  2.050  2.950  0.792  0.842
*4
  SPH     2.500  2.500  0.421  4.243
*5
  SPH     2.5    2.5    0.421  10.0
  END
*ZONES
  Z01 +1
  Z02 +2
  Z03 +3
  Z04 +4 -1 -2 -3
* ESCAPE ZONE IS A VOID SPHERE OF RADIUS  10 CM ENCLOSING THE SLAB
  Z05 +5  -1 -2 -3 -4
END
*MATERIAL
 1
 0
 2
 0
 0
*********************** SOURCE ********************
ELECTRONS
ENERGY  3.5
POSITION    2.5 2.5  -0.7071067
 RADIUS 1.0
DIRECTION 45.0  0.0
```

Figure 7. ACCEPT Input data file for 3.5 MeV 45° slant disk source incident on aluminum / void / silicon configuration shown in Figure 6.

2.3.2 Individual Electron Track Option

The second modification to ACCEPT permits the user to view the energy deposition contributions of individual case histories (electron tracks) in as many as 10 problem geometry cells. To implement this option, a line of the following form

"INDIVIDUAL-HISTS 92 145 93 146"

is added to the ACCEPT input file. The four numbers shown in the above example are the cell numbers corresponding to the electrically active parts (92, 145) of the CEASE front and back silicon detectors, respectively, and their corresponding electrically inactive parts (93, 146). A supplementary output file (EDSHOW.TXT) is produced by this version of ACCEPT and consists of: (a) tables (5 columns), for each cell, of (1) the case history number, energy deposition contributions [MeV] by the (2) primary electron, (3) knock-on electrons, (4) bremsstrahlung produced secondary electrons [see Figure 8], and (5) total energy deposition; and (b) a summary table showing the total energy deposition, for each history, in each cell (for example, 4 columns corresponding to cells 92, 145, 93, 146)[see Figure 9]. This last summary table allows for quick recognition of coincidence events occurring in the front and back detectors.

ENERGY DEPOSITION IN CELL NO. 92

Hist. no.	Primary	Knock-on	Secondary	Total
1	.44027E-01	.00000E+00	.00000E+00	.44027E-01
2	.52327E-01	.00000E+00	.00000E+00	.52327E-01
3	.77735E-01	.00000E+00	.00000E+00	.77735E-01
4	.11630E+00	.00000E+00	.00000E+00	.11630E+00
5	.53479E-01	.00000E+00	.00000E+00	.53479E-01
6	.61459E-01	.00000E+00	.00000E+00	.61459E-01
7	.10316E+00	.00000E+00	.00000E+00	.10316E+00
8	.53607E-01	.00000E+00	.00000E+00	.53607E-01
9	.49054E-01	.00000E+00	.00000E+00	.49054E-01
•				
•				
•				
•				
96	.44499E-01	.00000E+00	.00000E+00	.44499E-01
97	.76406E-01	.00000E+00	.00000E+00	.76406E-01
98	.71468E-01	.00000E+00	.00000E+00	.71468E-01
99	.88710E-01	.00000E+00	.00000E+00	.88710E-01
100	.56262E-01	.00000E+00	.00000E+00	.56262E-01

ENERGY DEPOSITION IN CELL NO. 145

Hist. no.	Primary	Knock-on	Secondary	Total
1	.45224E+00	.00000E+00	.00000E+00	.45224E+00
2	.00000E+00	.00000E+00	.00000E+00	.00000E+00
3	.00000E+00	.00000E+00	.00000E+00	.00000E+00
4	.38100E+00	.00000E+00	.00000E+00	.38100E+00
•				
•				
•				
•				
94	.00000E+00	.00000E+00	.00000E+00	.00000E+00
95	.00000E+00	.00000E+00	.00000E+00	.00000E+00
96	.35406E+00	.00000E+00	.00000E+00	.35406E+00
97	.00000E+00	.00000E+00	.00000E+00	.00000E+00
98	.18983E+00	.00000E+00	.00000E+00	.18983E+00
99	.40809E+00	.00000E+00	.00000E+00	.40809E+00
100	.00000E+00	.00000E+00	.00000E+00	.00000E+00

Figure 8. Portion of EDSHOW.TXT file produced by ACCEPT when the "INDIVIDUAL-HISTS" option is exercised. The table itemizes contributions to energy deposition attributable to primary, knock-on and secondary electrons for every case history.

TOTAL ENERGY DEPOSITION IN CELLS				
	92	145	93	146
Hist.no.				
1	.44027E-01	.45224E+00	.00000E+00	.00000E+00
2	.52327E-01	.00000E+00	.00000E+00	.00000E+00
3	.77735E-01	.00000E+00	.00000E+00	.00000E+00
4	.11630E+00	.38100E+00	.00000E+00	.00000E+00
5	.53479E-01	.44387E+00	.00000E+00	.00000E+00
6	.61459E-01	.00000E+00	.00000E+00	.00000E+00
7	.10316E+00	.00000E+00	.00000E+00	.00000E+00
8	.53607E-01	.16434E+00	.00000E+00	.00000E+00
9	.49054E-01	.44833E+00	.00000E+00	.00000E+00
			• • • •	
97	.76406E-01	.00000E+00	.00000E+00	.00000E+00
98	.71468E-01	.18983E+00	.00000E+00	.00000E+00
99	.88710E-01	.40809E+00	.00000E+00	.00000E+00
100	.56262E-01	.00000E+00	.00000E+00	.00000E+00

Figure 9. Portion of EDSHOW.TXT file produced by ACCEPT when the "INDIVIDUAL-HISTS" option is exercised. This table lists total energy deposition in all requested cells for every case history.

Program listings of the ACCEPT subroutines that were modified to incorporate the "RECTANGLE-SOURCE", "CIRCLE-SOURCE" and "INDIVIDUAL-HISTS" options are given in Appendix 2.

3. Proton Transport Modeling

3.1 Energy Deposition Calculations - CEASE Telescope

Energy deposition plots (see Figure 10) for electrons and protons were supplied by AFRL [6]. The curves shown in Figure 10 represent calculations for the front (DFT - thickness = 0.015 cm) and back (DBT - thickness = 0.05cm) silicon wafer detectors in the CEASE telescope. This provided an opportunity to test the validity of our CEASE model and the physical realism of MCNPX proton transport calculations. The CEASE model [7] written earlier for MCNPX was used in eight Monte Carlo runs corresponding to proton source energies, 4, 4.5, 5, 7.5, 9.5,15, 30, 100 MeV with normally incident protons (disk sources covering the telescope aperture). The MCNPX simulation results, $Edep_{DBT}$ vs. $Edep_{DFT}$ for the proton sources, are plotted in Figure 11. They appear to closely track the proton energy deposition curves of Figure 10.

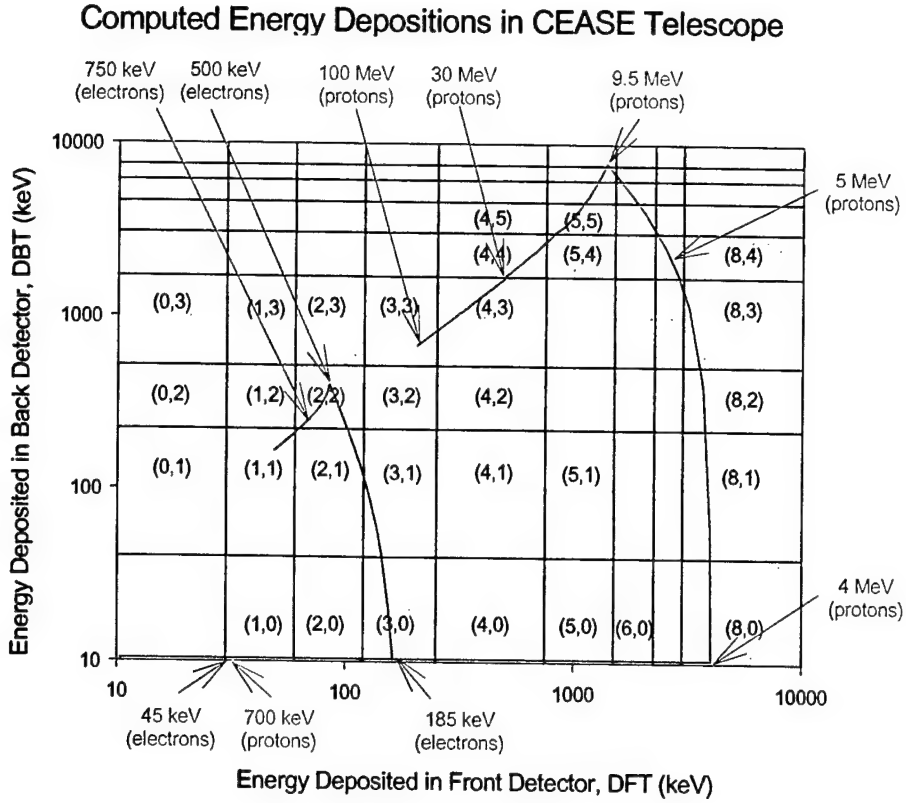


Figure 10. Computed energy depositions due to protons and electrons in DBT vs. DFT for the CEASE telescope[4,6]

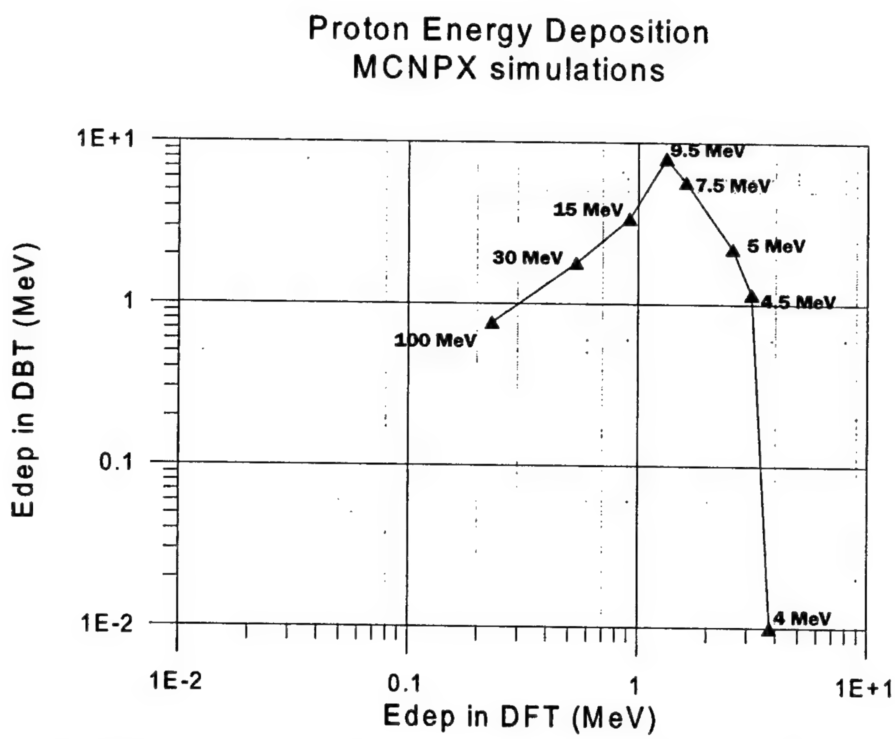


Figure 11. Energy depositions due to protons in DBT vs. DFT for the CEASE telescope as computed with the MCNPX simulation program.

3.2 Coincidence Event Identification - CEASE

A short FORTRAN program, count.F, was written to analyze the track file ("ptrac") produced by MCNPX in order to identify coincidence events in the DFT and DBT detectors. This program lists the energy deposition from protons in each detector for each proton track history and enables the investigator to recognize and evaluate coincidence events. The program is also configured to record the energy deposition in any cell of the CEASE simulation geometry. An annotated sample output file displaying the accounting results for 10000 proton histories (100 MeV proton source in CEASE aperture, normal incidence) in shown in Figure 12. The cell numbers containing the coincidence events listed at the end of the output identify tracks of interest that can be re-examined. The program listing of count.F is given in Appendix 3.

MCNPX - CEASE cell #	Total energy deposition	# of protons in cell	energy deposition due to protons	# of electrons in cell	energy deposition due to electrons	# of neutrons in cell	Energy deposition due to neutrons	# of photons in cell	Energy deposition due to photons
1	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
2	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
3	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
					•				
					•				
24	0.47660E-02	13	0.47660E-02	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
25	0.14176E+00	125	0.14176E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
26	0.31609E+00	246	0.31603E+00	0	0.00000E+00	0	0.00000E+00	1	0.58880E-04
27	0.46626E+00	357	0.46478E+00	0	0.00000E+00	0	0.00000E+00	1	0.14821E-02
28	0.60600E+00	467	0.60600E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
29	0.73127E+00	549	0.73111E+00	0	0.00000E+00	0	0.00000E+00	1	0.16188E-03
30	0.83742E+00	631	0.83742E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
31	0.96039E+00	704	0.95784E+00	0	0.00000E+00	0	0.00000E+00	2	0.25468E-02
32	0.10499E+01	760	0.10499E+01	0	0.00000E+00	0	0.00000E+00	1	0.45840E-04
33	0.11213E+01	807	0.11213E+01	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
34	0.12038E+01	874	0.12038E+01	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
35	0.12848E+01	906	0.12848E+01	0	0.00000E+00	0	0.00000E+00	1	0.21850E-04
36	0.13170E+01	931	0.13170E+01	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
37	0.15761E-01	290	0.15761E-01	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
38	0.94740E+00	333	0.94740E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
39	0.12217E-01	705	0.12217E-01	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
40	0.28122E+01	700	0.28122E+01	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
41	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
42	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
43	0.17045E+00	229	0.17045E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
44	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
45	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
46	0.64099E+00	605	0.64099E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
47	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
48	0.26440E-02	0	0.00000E+00	0	0.00000E+00	1	0.13955E-05	3	0.26426E-02
49	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
50	0.26963E+01	198	0.26587E+01	0	0.00000E+00	6	0.63279E-02	31	0.31316E-01
51	0.75840E+00	228	0.75837E+00	0	0.00000E+00	1	0.12650E-04	1	0.11150E-04
					•				
					•				
61	0.94892E+00	667	0.94866E+00	0	0.00000E+00	0	0.00000E+00	2	0.26012E-03
62	0.20318E+00	684	0.20318E+01	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00
63	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00	0	0.00000E+00

Figure 12. Output file from count.F interpreting particle track, coincidence event and energy deposition data from MCNPX-ptrac file for protons, electrons, neutrons and photons.

```
 68   0.00000E+00    0   0.00000E+00   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
 69   0.15190E+00  523   0.15190E+00   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
 70   0.18603E+00  209   0.18603E+00   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
 71   0.00000E+00    0   0.00000E+00   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
 72   0.34538E+00  142   0.34538E+00   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
 73   0.00000E+00    0   0.00000E+00   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
 74   0.14732E+01  581   0.14732E+01   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
 75   0.26610E+01  671   0.26585E+01   0   0.00000E+00   0   0.00000E+00   1   0.25213E-02
 76   0.22022E+01  606   0.22017E+01   0   0.00000E+00   0   0.00000E+00   1   0.45914E-03
 77   0.00000E+00    0   0.00000E+00   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
 78   0.79859E+00  657   0.79859E+00   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
 79   0.00000E+00    0   0.00000E+00   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
 80   0.00000E+00    0   0.00000E+00   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
                                              •
                                              •
128   0.12088E+01  238   0.12080E+01   0   0.00000E+00   1   0.45992E-05   1   0.80680E-03
129   0.10351E+01  209   0.10351E+01   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00
130   0.88508E+00  180   0.88487E+00   0   0.00000E+00   1   0.11277E-04   3   0.20188E-03
131   0.88338E+00  164   0.88333E+00   0   0.00000E+00   1   0.37793E-04   1   0.87200E-05
                                              •
                                              •
146   0.45599E-02    1   0.45599E-02   0   0.00000E+00   0   0.00000E+00   0   0.00000E+00

                             photons                                  protons
                neutrons
no. of escaped particles =   114   140  0  0  0  0  0  0   145  0
 escaped energy =    0.59827E+01
total energy deposited =     0.93343E+02
   from protons =     0.90556E+02
   from photons =     0.69015E-01
   from neutrons =     0.24006E-01
   from electrons =     0.00000E+00
   from inelastic collisions =    0.26944E+01
   number of proton coincidence events =  272
   number of neutron coincidence events =    4
   number of photon coincidence events =    2

history numbers for proton coincidence events
   2    3    4    5    7   14   16   23   27   30   35   40   42   45   48
  51   54   61   64   69   72   73   79   83   86   88   89   93   94   95
  99  103  106  111  114  115  117  121  125  129  134  137  138  147  156
 161  169  172  176  177  181  183  185  188  192  200  201  205  212  217
 219  220  221  222  226  230  232  235  236  240  245  247  255  259  264
 269  271  276  281  282  285  290  295  296  302  306  310  314  316  320
 325  326  328  334  338  341  342  347  348  354  357  358  369  373  374
 375  376  384  388  396  398  400  402  407  408  409  412  418  421  422
 426  427  430  433  436  437  440  446  455  460  467  468  472  475  480
 483  485  489  491  496  498  500  503  505  509  511  512  521  523  524
 533  537  541  542  551  557  558  565  569  571  576  578  582  585  586
 589  591  592  596  601  602  605  614  618  619  629  632  635  642  645
 656  659  662  663  664  669  670  672  676  679  681  684  687  690  693
 697  701  705  707  709  712  717  721  722  729  734  746  753  759  762
 767  768  776  779  786  788  793  796  799  801  804  807  809  813  816
 819  824  828  833  841  843  845  847  850  853  854  859  863  865  867
 869  871  874  876  885  886  888  889  892  901  904  906  913  916  921
 925  931  934  936  937  939  943  946  961  963  964  970  973  979  980
 983  997
```

Figure 12 (cont.). Output file from count.F interpreting particle track, coincidence event and energy deposition data from MCNPX-ptrac file for protons, electrons, neutrons and photons.

3.3 Beam Source Subroutine for MCNPX

A specialized source subroutine that allows the user to run beam sources at arbitrary positions and arbitrary orientations was written for use with MCNPX. MCNPX makes allowance for user-supplied source routines in addition to those supplied in the program. This is accomplished by omission of the source descriptor records in the MCNPX input data file. When this is done, the programs searches for the user-supplied "SUBROUTINE SOURCE". In our version the user enters, as input to screen prompts, the source position (x, y, z), source direction cosines (u, v, w), source particle type (ipt), source particle energy (MeV), geometry surface number on which or cell number in which the source origination point is located. The program listing for source.F is given in Appendix 4.

4. Dome Dosimeter Study

Electron and proton transport calculations were made for the CEASE DD1 and DD2 dosimeters [4] and PASP Dome D2 and D3 dosimeters [8,9] using the ITS/ACCEPT and MCNPX simulation programs. The purpose of these calculations was to provide a means for determining the effects of differences in shield geometry on dose measurements in the same radiation environment. The data sets obtained with the CEASE and PASP dosimeters, normalized to take into account differences in shielding geometry, would then be used to study the solar cycle dependence of the electron dose from the outer radiation belt.

The schematic shown in Figure 13 for the CEASE DD1 and DD2 dosimeter assemblies was provided by AFRL[10]. The dosimeter assembly consists of a flat rectangular silicon diode resting on an aluminum oxide substrate, which in turn is mounted on an aluminum base. The dosimeter is capped with an aluminum plate.

4.1 CEASE Dosimeter Models - ACCEPT and MCNPX

Models for the original CEASE DD1 and DD2 geometries were obtained directly using the "CIRCLE-SOURCE" option as described in Section 2.3.1, above. Isotropic electron sources were assumed uniformly distributed on the disc surface as shown in Figure 14. We then wrote and installed a new source option, "DOME SOURCE", for ITS/ACCEPT in which the isotropic, inward-directed electron source is assumed to be uniformly distributed on the surface of a thin void hemispherical shell with the same radius as the disc shield plates. This source is depicted in the ACCEPT geometry schematic of Figure 15. The ACCEPT input file corresponding to the geometries shown in Figures 14 and 15 is listed in Appendix 5 for the CEASE DD1 dosimeter. Both the "CIRCLE-SOURCE" and "DOME SOURCE" input data (see annotations) are shown in the same file listing to conserve space.

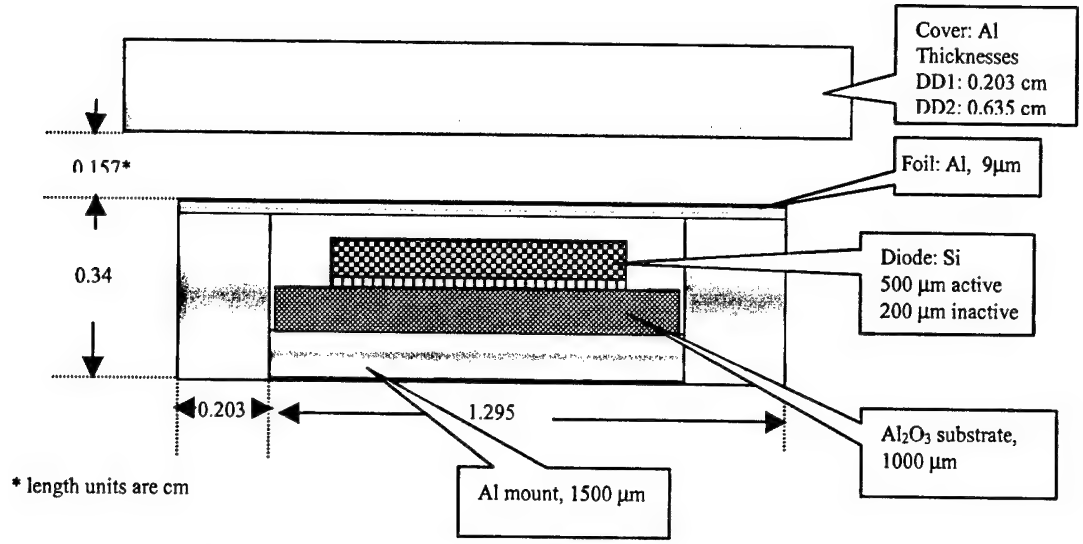


Figure 13. CEASE DD1 and DD2 dosimeter assemblies [10].

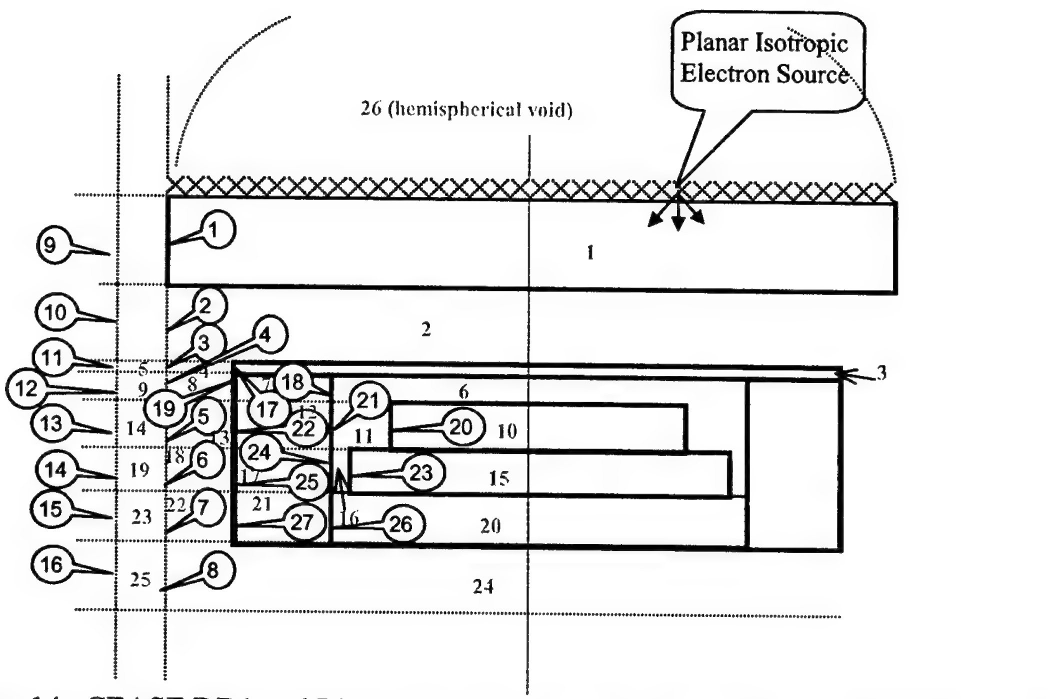


Figure 14. CEASE DD1 and DD2 dosimeter geometry schematic for ACCEPT showing plane isotropic electron source; ◯= geometry zones; nn = material cells.

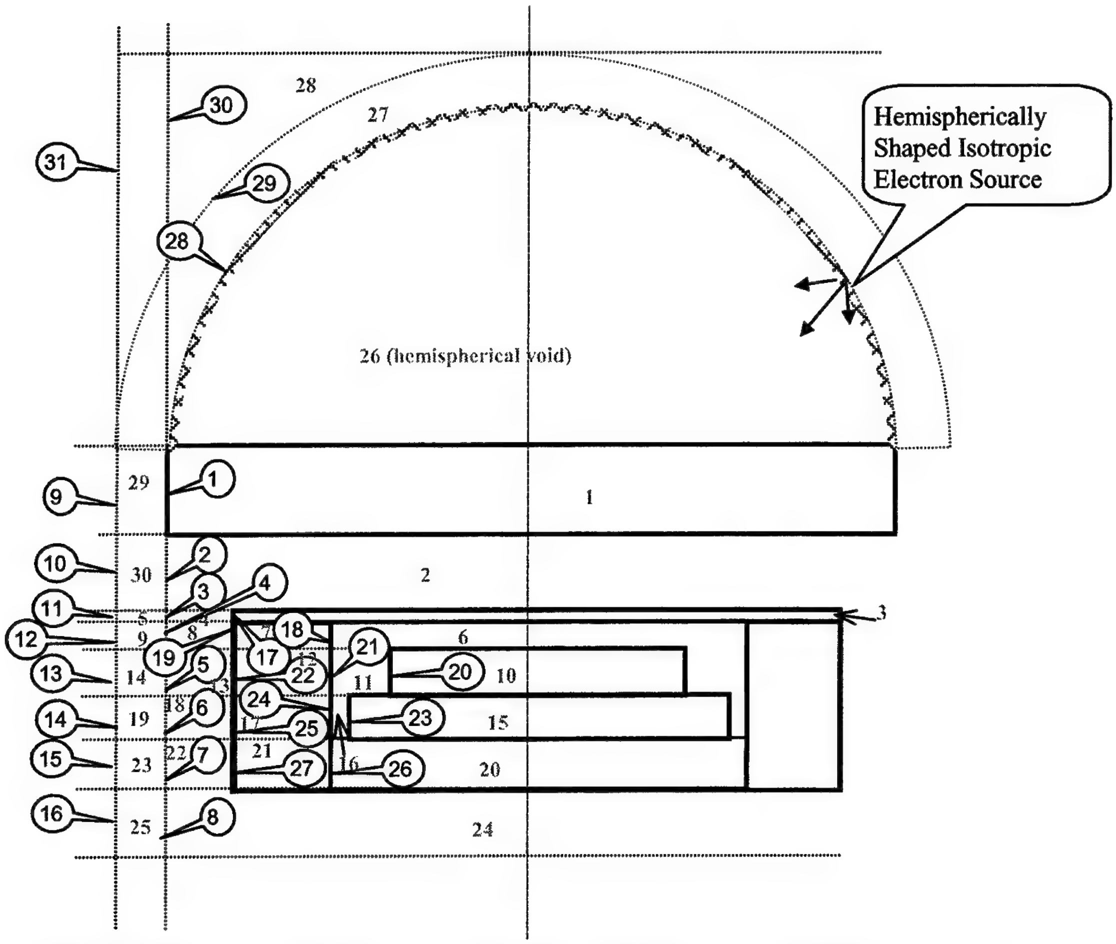


Figure 15. CEASE DD1 and DD2 dosimeter geometry schematic for ACCEPT showing hemispherically shaped isotropic electron source; ◯ = geometry zones; nn = material cells.

The equivalent simulation scenarios were executed using MCNPX. Figure 16 is a surface and cell schematic for the DD1, DD2 MCNPX model corresponding to that shown for ACCEPT in Figure 15. The corresponding MCNPX input file for DD2 is given in Appendix 6. For the case of the flat plate source, the MCNPX-supplied source provided a cosine-isotropic source option that could be implemented using the run input data file. It was necessary to write a source subroutine for MCNPX that allowed isotropy in angle rather than cosine. It was also necessary to write new source subroutines for the hemispherical shell sources. These code modifications provided, along with appropriate geometry factor adjustments, a tool for the AFRL researchers to compare simulated dosimeter responses with different source geometry assumptions.

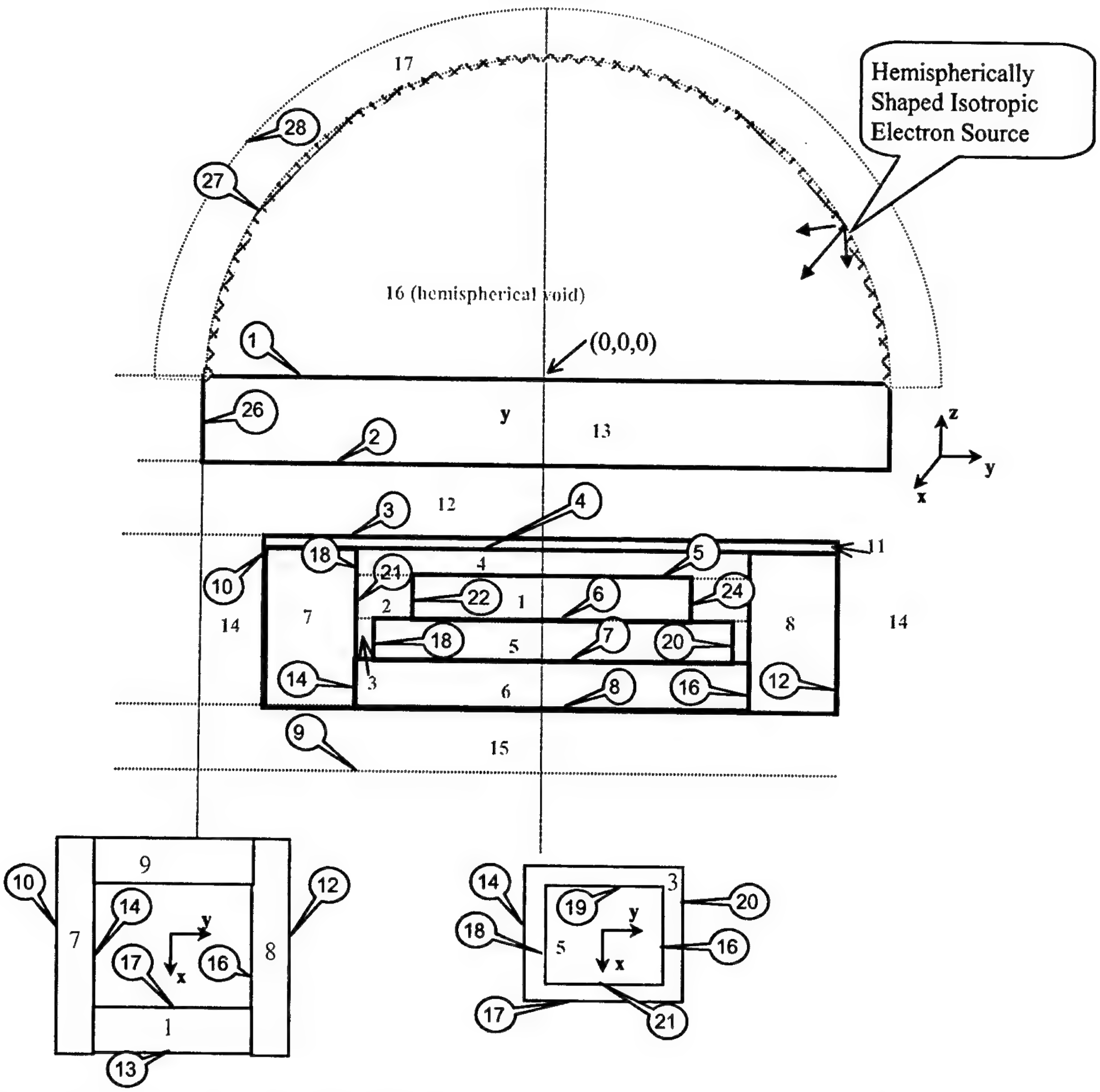


Figure 16. CEASE DD1 and DD2 dosimeter geometry schematic for MCNPX showing hemispherically shaped isotropic electron source; ◯ = surfaces; nn = material cells.

4.2 PASP Dosimeter Models - ACCEPT and MCNPX

The PASP dome dosimeters were modeled using the "HEMISPHERICAL DOME SOURCE" option in ITS/ACCEPT and new source routines for MCNPX. The ACCEPT geometry schematics and source configuration for the PASP Dome 2 and Dome 3 dosimeters are shown in Figures 17a and 17b, respectively. The corresponding geometry schematics for MCNPX are shown in Figures 18a and 18b. The computer code listings for the "DOME SOURCE" option in ACCEPT and the specialized source subroutines for MCNPX are given in Appendices 7 and 8, respectively.

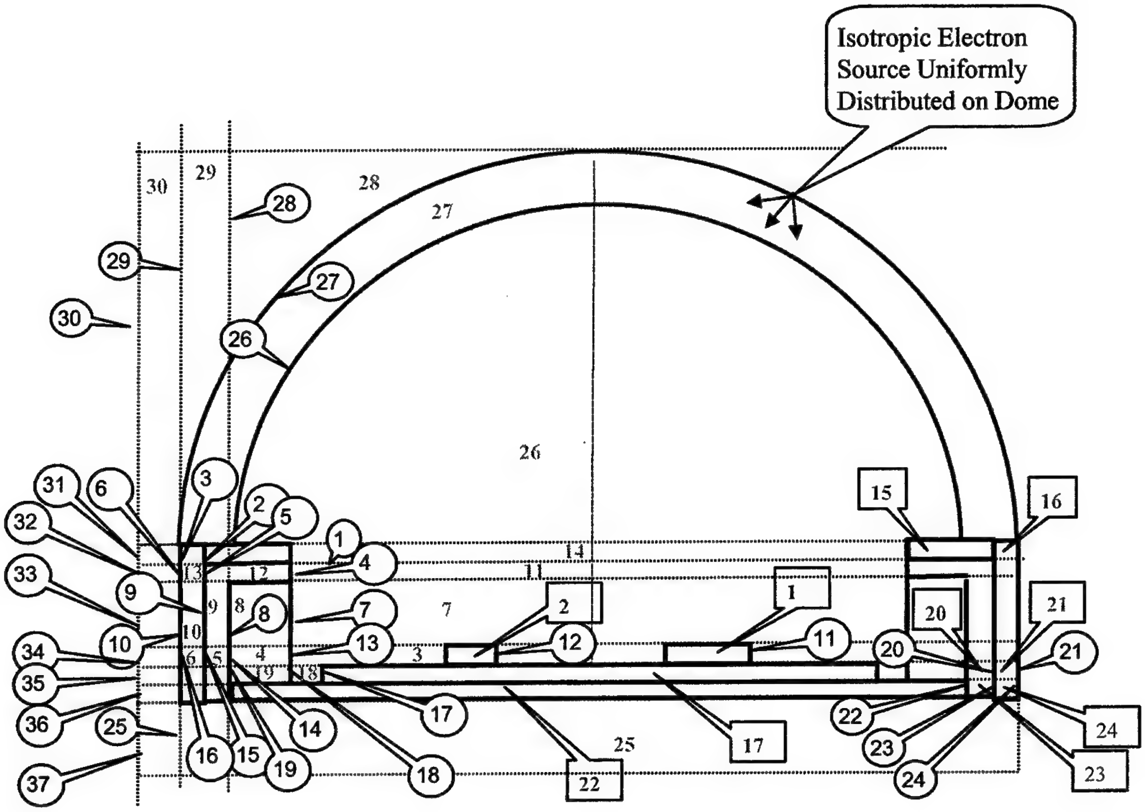


Figure 17a. "PASP Dome 2" dosimeter[8,9] geometry Schematic for ACCEPT showing isotropic electron source incident on surface of Al dome; = geometry zones; nn = material cells.

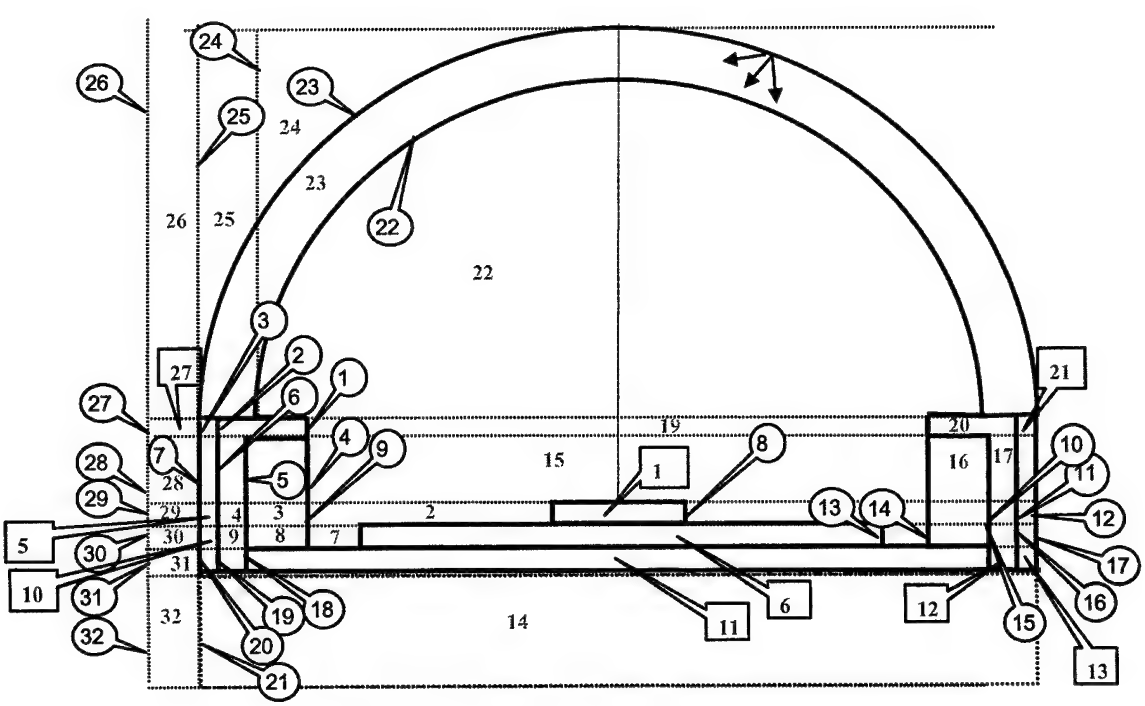


Figure 17b. "PASP Dome 3" dosimeter[8,9] geometry schematic for ACCEPT showing isotropic electron source incident on surface of Al dome; = geometry zones; nn = material cells.

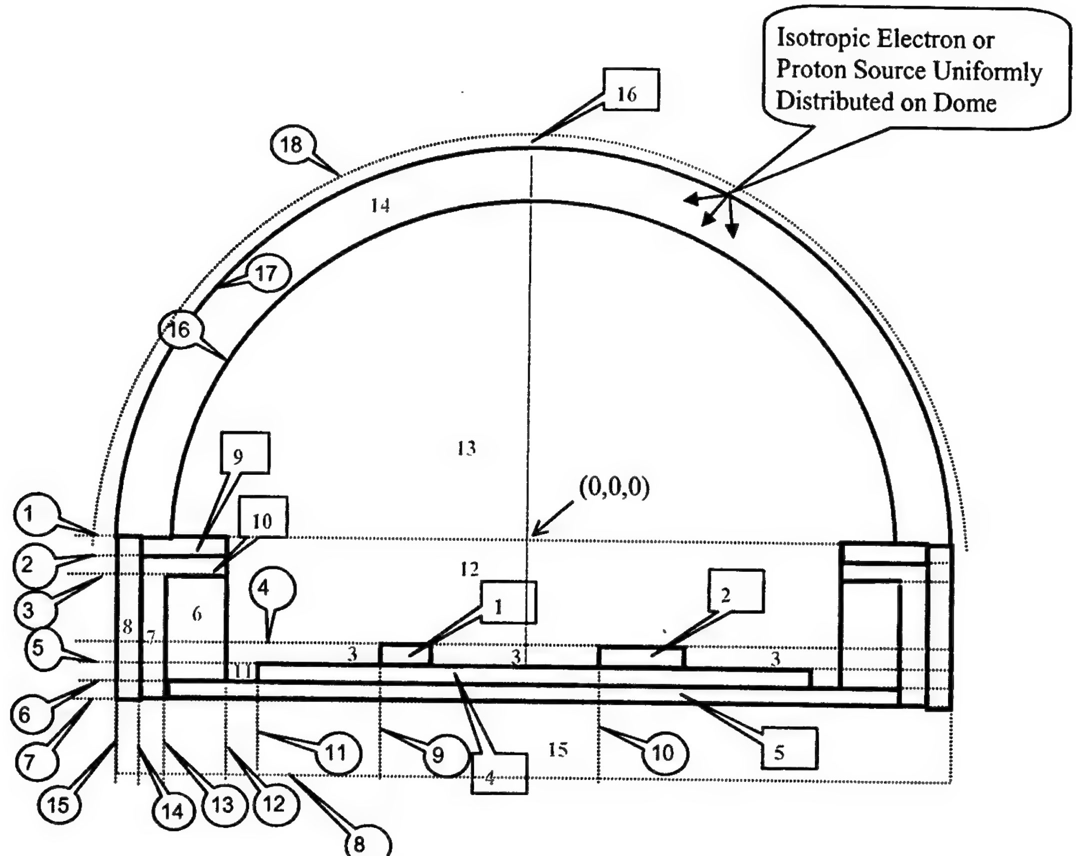


Figure 18a. "PASP Dome 2" dosimeter geometry schematic for MCNPX showing isotropic electron or proton source incident on surface of Al dome; = surfaces; nn = material cells.

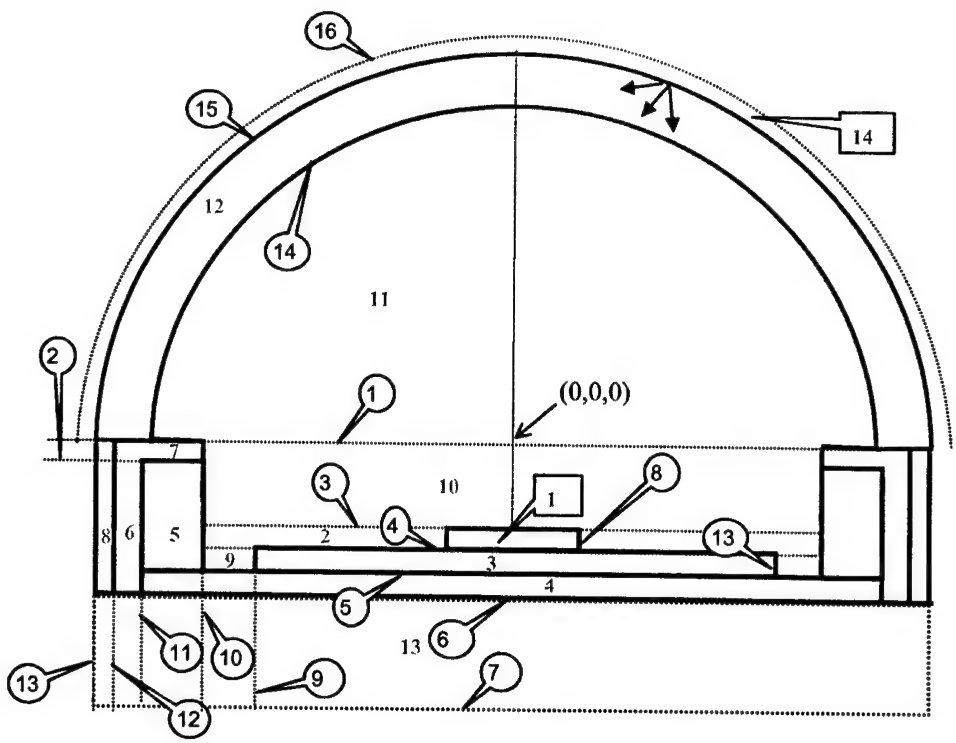


Figure 18b. "PASP Dome 3" dosimeter geometry schematic for MCNPX showing isotropic electron or proton source incident on surface of Al dome; = surfaces; nn = material cells.

In all cases, trial runs were made with small numbers of histories (10000) to check on equivalence of the results obtained with both codes. While the modified version of ITS/ACCEPT was sufficient to accomplish the immediate goals of modeling electron transport, the addition of MCNPX provided: (1) a check on the ITS/ACCEPT results; and (2) a capability for modeling proton transport with the same problem geometries. Electron transport simulations were run at AFRL using the CEASE and PASP dosimeter models described above for several power law electron energy spectra characteristic of the outer belt electron environment. Results are reported in [11].

5. Summary

During the period covered by this report, the technical activity and progress achieved consisted primarily of: (1) modeling of electron transport and calculation of electron energy deposition in silicon dosimeter wafers; (2) construction of a highly detailed ITS-ACCEPT computer model for the HEP sensor, in-flight model; (3) design and installation of enhancements such as expanded source geometry repertoire, single history tracking, and coincidence event recognition capability, to the ITS-ACCEPT transport program; (4) construction of geometry and electron and proton source models for CEASE and PASP dosimeter studies; (5) acquisition and implementation of a three-dimensional geometry construction and visualization program that is compatible with both the ITS and MCNP code series; (6) providing assistance, advice, input data files and computer code enhancements to AFRL for implementation by AFRL research personnel.

We anticipate continuing this and related research efforts by providing simulation calculations and results, computer code enhancements and new geometry models, and by performing in an advisory capacity to the sponsor.

References

1. *ITS - Integrated TIGER Series of Coupled Electron /Photon Monte Carlo Code System*, J. A. Halbleib *et al.* ORNL RSICC Computer Code Package CCC-467.
2. *MCNPX ™ , Version 2.1.5 User's Manual*, L. S. Waters, Ed., Los Alamos Radiation Transport Group(X-6), November 14,1999.
3. *Sabrina 4.15 for Windows*, Copyright 2001, White Rock Science, P.O. Box 4729, White Rock, NM, 87544
4. B. Dichter, *et al.*, Compact Environmental Anomaly Sensor (CEASE): A Novel Spacecraft Instrument for *In Situ* Measurements of Environmental Conditions, *IEEE Trans. Nucl. Sci.* **45(6)**, 2758, Dec. 1998.
5. R. Redus, HEP Flight Model, June 30, 1998, Amptek, Inc., Bedford, MA.
6. D. Brautigam, AFRL/VSBXR, private communication, October 4, 2000.
7. S. Woolf, Installation and Operation of Particle Transport Simulation Programs to Model the Detection and Measurement of Space Radiation by Space-borne Sensors, Air Force Research Laboratory Report AFRL-VS-TR-2001-1605, December 29, 2000.
8. M. S. Gussenhoven, *et al.*, Low altitude orbit edge of the inner radiation belt: Dose models from the APEX satellite, *IEEE Trans. Nucl. Sci.* **42(6)**, 2035, December1995.
9. M. S. Gussenhoven, *et al.*, Low altitude orbit dose as a function of inclination, magnetic activity and solar cycle, *IEEE Trans. Nucl. Sci.* **44(6)**, 2161, December1997.
10. D. Brautigam, AFRL/VSBXR, private communication, May 15, 2001.
11. D. Brautigam *et al.*, Solar cycle variation of outer belt electron dose at low earth orbit, to appear in *IEEE Trans. Nucl. Sci.* **48(6)**, Dec. 2001.

APPENDIX 1

Annotated ITS-ACCEPT Input Data File for the HEP In-Flight Instrument

```
 25 MEV ELECTRON DISK SOURCE - HEP FLIGHT MODEL, FRONT ENTRY, NORMAL INCIDENCE
********************** GEOMETRY **********************
GEOMETRY
*BODIES
*1
  TRC   0.0   0.0   0.0       0.0   0.00000  0.25400        0.6096 0.562396
*2
  RCC   0.0   0.0   0.0       0.0   0.00000  0.25400        0.80010
*3
  RCC   0.0   0.0   0.0       0.0   0.00000  0.25400        1.58877
*4
  RCC   0.0   0.0   0.0       0.0   0.00000  0.25400        2.0955
*5
  TRC   0.0   0.0   0.25400   0.0   0.00000  0.25400        0.562396 0.5151928
*6
  RCC   0.0   0.0   0.25400   0.0   0.00000  0.25400        0.80010
*7
  RCC   0.0   0.0   0.25400   0.0   0.00000  0.25400        2.0955
*8
  TRC   0.0   0.0   0.50800   0.0   0.00000  0.9271         0.5151928 0.3429
*9
  RCC   0.0   0.0   0.50800   0.0   0.00000  0.9271         0.80010
*10
  RCC   0.0   0.0   0.50800   0.0   0.0      0.9271         1.58877
*11
  RCC   0.0   0.0   0.50800   0.0   0.00000  0.9271         2.0955
*12
  RCC   0.0   0.0   1.43510   0.0   0.00000  0.11690        1.58877
*13
  RCC   0.0   0.0   1.43510   0.0   0.00000  0.11690        2.0955
*14
  RCC   0.0   0.0   1.55200   0.0   0.00000  0.05070        1.58877
*15
  RCC   0.0   0.0   1.55200   0.0   0.00000  0.05070        2.0955
*16
  RCC   0.0   0.0   1.60270   0.0   0.00000  0.01274        0.47625
*17
  RCC   0.0   0.0   1.60270   0.0   0.00000  0.01274        1.43510
*18
  RCC   0.0   0.0   1.60270   0.0   0.00000  0.01274        1.58877
*19
  RCC   0.0   0.0   1.60270   0.0   0.00000  0.01274        2.0955
*20
  TRC   0.0   0.0   1.61544   0.0   0.0      0.08128        0.5      0.38000
*21
  RCC   0.0   0.0   1.61544   0.0   0.0      0.08128        0.50000
*22
  RCC   0.0   0.0   1.61544   0.0   0.0      0.08128        0.69850
*23
  RCC   0.0   0.0   1.61544   0.0   0.0      0.08128        1.43510
*24
  RCC   0.0   0.0   1.61544   0.0   0.0      0.08128        1.58877
*25
  RCC   0.0   0.0   1.61544   0.0   0.0      0.08128        2.0955
*26
  TRC   0.0   0.0   1.69672   0.0   0.0      0.06668        0.38000 0.28209
*27
  RCC   0.0   0.0   1.69672   0.0   0.00000  0.06668        0.5
*28
  RCC   0.0   0.0   1.69672   0.0   0.00000  0.06668        0.6350
*29
  RCC   0.0   0.0   1.69672   0.0   0.00000  0.06668        0.6985
*30
  RCC   0.0   0.0   1.69672   0.0   0.00000  0.06668        1.43510
*31
  RCC   0.0   0.0   1.69672   0.0   0.00000  0.06668        1.58877
```

```
*32
 RCC   0.0   0.0   1.69672   0.0   0.00000  0.06668   2.0955
*33
 RCC   0.0   0.0   1.76340   0.0   0.00000  0.02600   0.28209
*34
 RCC   0.0   0.0   1.76340   0.0   0.00000  0.02600   0.30000
*35
 RCC   0.0   0.0   1.76340   0.0   0.00000  0.02600   0.50000
*36
 RCC   0.0   0.0   1.76340   0.0   0.00000  0.02600   0.63500
*37
 RCC   0.0   0.0   1.76340   0.0   0.00000  0.02600   0.69850
*38
 RCC   0.0   0.0   1.76340   0.0   0.00000  0.02600   1.43510
*39
 RCC   0.0   0.0   1.76340   0.0   0.00000  0.02600   1.58877
*40
 RCC   0.0   0.0   1.76340   0.0   0.00000  0.02600   2.09550
*41
 RCC   0.0   0.0   1.78940   0.0   0.00000  0.00801   0.28209
*42
 RCC   0.0   0.0   1.78940   0.0   0.00000  0.00801   0.48200
*43
 RCC   0.0   0.0   1.78940   0.0   0.00000  0.00801   0.50000
*44
 RCC   0.0   0.0   1.78940   0.0   0.00000  0.00801   0.63500
*45
 RCC   0.0   0.0   1.78940   0.0   0.00000  0.00801   0.69850
*46
 RCC   0.0   0.0   1.78940   0.0   0.00000  0.00801   1.43510
*47
 RCC   0.0   0.0   1.78940   0.0   0.00000  0.00801   1.58877
*48
 RCC   0.0   0.0   1.78940   0.0   0.00000  0.00801   2.09550
*49
 RCC   0.0   0.0   1.79741   0.0   0.0      0.03400   0.48200
*50
 RCC   0.0   0.0   1.79741   0.0   0.0      0.03400   0.50000
*51
 RCC   0.0   0.0   1.79741   0.0   0.0      0.03400   0.69850
*52
 RCC   0.0   0.0   1.79741   0.0   0.0      0.03400   1.43510
*53
 RCC   0.0   0.0   1.79741   0.0   0.0      0.03400   1.58877
*54
 RCC   0.0   0.0   1.79741   0.0   0.0      0.03400   2.09550
*55
 RCC  0.0    0.0   1.83141   0.0   0.0      0.00801   0.48200
*56
 RCC  0.0    0.0   1.83141   0.0   0.0      0.00801   0.50000
*57
 RCC  0.0    0.0   1.83141   0.0   0.0      0.00801   0.69850
*58
 RCC  0.0    0.0   1.83141   0.0   0.0      0.00801   1.43510
*59
 RCC  0.0    0.0   1.83141   0.0   0.0      0.00801   1.58877
*60
 RCC  0.0    0.0   1.83141   0.0   0.0      0.00801   2.09550
*61
 RCC  0.0    0.0   1.83942   0.0   0.0      0.14432   0.50000
*62
 RCC  0.0    0.0   1.83942   0.0   0.0      0.14432   0.69850
*63
 RCC  0.0    0.0   1.83942   0.0   0.0      0.14432   1.43510
*64
 RCC  0.0    0.0   1.83942   0.0   0.0      0.14432   1.58877
*65
 RCC  0.0    0.0   1.83942   0.0   0.0      0.14432   2.09550
*66
 RCC  0.0    0.0   1.98374   0.0   0.0      0.01270   0.47625
*67
```

```
 RCC  0.0    0.0   1.98374   0.0   0.0      0.01270    1.43510
*68
 RCC  0.0    0.0   1.98374   0.0   0.0      0.01270    1.58877
*69
 RCC  0.0    0.0   1.98374   0.0   0.0      0.01270    2.09550
*70
 RCC  0.0    0.0   1.99644   0.0   0.0      0.03810    1.58877
*71
 RCC  0.0    0.0   1.99644   0.0   0.0      0.03810    2.09550
* 72
 RCC      .0     .0    2.03454    .0     .0  .0507000 1.5887700
* 73
 RCC      .0     .0    2.03454    .0     .0  .0507000 2.0955000
* 74
 RCC      .0     .0    2.08524    .0     .0  .0127400  .4762500
* 75
 RCC      .0     .0    2.08524    .0     .0  .0127400 1.4351000
* 76
 RCC      .0     .0    2.08524    .0     .0  .0127400 1.5887700
* 77
 RCC      .0     .0    2.08524    .0     .0  .0127400 2.0955000
* 78
 TRC      .0     .0    2.09798    .0     .0  .0812800  .5000000  .3800000
* 79
 RCC      .0     .0    2.09798    .0     .0  .0812800  .5000000
* 80
 RCC      .0     .0    2.09798    .0     .0  .0812800  .6985000
* 81
 RCC      .0     .0    2.09798    .0     .0  .0812800 1.4351000
* 82
 RCC      .0     .0    2.09798    .0     .0  .0812800 1.5887700
* 83
 RCC      .0     .0    2.09798    .0     .0  .0812800 2.0955000
* 84
 TRC      .0     .0    2.17926    .0     .0  .0666800  .3800000  .2820900
* 85
 RCC      .0     .0    2.17926    .0     .0  .0666800  .5000000
* 86
 RCC      .0     .0    2.17926    .0     .0  .0666800  .6350000
* 87
 RCC      .0     .0    2.17926    .0     .0  .0666800  .6985000
* 88
 RCC      .0     .0    2.17926    .0     .0  .0666800 1.4351000
* 89
 RCC      .0     .0    2.17926    .0     .0  .0666800 1.5887700
* 90
 RCC      .0     .0    2.17926    .0     .0  .0666800 2.0955000
* 91
 RCC      .0     .0    2.24594    .0     .0  .0260000  .2820900
* 92
 RCC      .0     .0    2.24594    .0     .0  .0260000  .3000000
* 93
 RCC      .0     .0    2.24594    .0     .0  .0260000  .5000000
* 94
 RCC      .0     .0    2.24594    .0     .0  .0260000  .6350000
* 95
 RCC      .0     .0    2.24594    .0     .0  .0260000  .6985000
* 96
 RCC      .0     .0    2.24594    .0     .0  .0260000   1.4351000
* 97
 RCC      .0     .0    2.24594    .0     .0  .0260000   1.5887700
* 98
 RCC      .0     .0    2.24594    .0     .0  .0260000   2.0955000
* 99
 RCC      .0     .0    2.27194    .0     .0  .0080100  .2820900
*100
 RCC      .0     .0    2.27194    .0     .0  .0080100  .4820000
*101
 RCC      .0     .0    2.27194    .0     .0  .0080100  .5000000
*102
 RCC      .0     .0    2.27194    .0     .0  .0080100  .6350000
```

```
*103
   RCC    .0    .0    2.27194    .0    .0  .0080100   .6985000
*104
   RCC    .0    .0    2.27194    .0    .0  .0080100   1.4351000
*105
   RCC    .0    .0    2.27194    .0    .0  .0080100   1.5887700
*106
   RCC    .0    .0    2.27194    .0    .0  .0080100   2.0955000
*107
   RCC    .0    .0    2.27995    .0    .0  .0340000   .4820000
*108
   RCC    .0    .0    2.27995    .0    .0  .0340000   .5000000
*109
   RCC    .0    .0    2.27995    .0    .0  .0340000   .6985000
*110
   RCC    .0    .0    2.27995    .0    .0  .0340000   1.4351000
*111
   RCC    .0    .0    2.27995    .0    .0  .0340000   1.5887700
*112
   RCC    .0    .0    2.27995    .0    .0  .0340000   2.0955000
*113
   RCC    .0    .0    2.31395    .0    .0  .0080100   .4820000
*114
   RCC    .0    .0    2.31395    .0    .0  .0080100   .5000000
*115
   RCC    .0    .0    2.31395    .0    .0  .0080100   .6985000
*116
   RCC    .0    .0    2.31395    .0    .0  .0080100   1.4351000
*117
   RCC    .0    .0    2.31395    .0    .0  .0080100   1.5887700
*118
   RCC    .0    .0    2.31395    .0    .0  .0080100   2.0955000
*119
   RCC    .0    .0    2.32196    .0    .0  .1443200   .5000000
*120
   RCC    .0    .0    2.32196    .0    .0  .1443200   .6985000
*121
   RCC    .0    .0    2.32196    .0    .0  .1443200   1.4351000
*122
   RCC    .0    .0    2.32196    .0    .0  .1443200   1.5887700
*123
   RCC    .0    .0    2.32196    .0    .0  .1443200   2.0955000
*124
   RCC    .0    .0    2.46628    .0    .0  .0127000   .4762500
*125
   RCC    .0    .0    2.46628    .0    .0  .0127000   1.4351000
*126
   RCC    .0    .0    2.46628    .0    .0  .0127000   1.5887700
*127
   RCC    .0    .0    2.46628    .0    .0  .0127000   2.0955000
*128
   RCC    .0    .0    2.47898    .0    .0  .0507600   1.5887700
*129
   RCC    .0    .0    2.47898    .0    .0  .0507600   2.0955000
*130
   RCC    .0    .0    2.52974    .0    .0  .0508400   0.75057
*131
   RCC    .0    .0    2.52974    .0    .0  .0508400   0.94615
*132
   RCC    .0    .0    2.52974    .0    .0  .0508400   1.45415
*133
   RCC    .0    .0    2.52974    .0    .0  .0508400   1.58877
*134
   RCC    .0    .0    2.52974    .0    .0  .0508400   2.0955
*135
   RCC    .0    .0    2.58058    .0    .0  3.0        0.75057
*136
   RCC    .0    .0    2.58058    .0    .0  3.0        0.94615
*137
   RCC    .0    .0    2.58058    .0    .0  3.0        1.45415
*138
```

```
 RCC    .0    .0   2.58058    .0    .0  3.0         1.58877
*139
 RCC    .0    .0   2.58058    .0    .0  3.0         2.0955
*140
 RCC    .0    .0   5.58058    .0    .0  0.05060     0.75057
*141
 RCC    .0    .0   5.58058    .0    .0  0.05060     0.94615
*142
 RCC    .0    .0   5.58058    .0    .0  0.05060     1.45415
*143
 RCC    .0    .0   5.58058    .0    .0  0.05060     1.58877
*144
 RCC    .0    .0   5.58058    .0    .0  0.05060     2.0955
*145
 RCC    .0    .0   5.63118    .0    .0  0.05080     1.58877
*146
 RCC    .0    .0   5.63118    .0    .0  0.05080     2.0955
*147
 RCC    .0    .0   5.68198    .0    .0  0.01270     1.4351
*148
 RCC    .0    .0   5.68198    .0    .0  0.01270     1.58877
*149
 RCC    .0    .0   5.68198    .0    .0  0.01270     2.0955
*150
 TRC    .0    .0   5.69468    .0    .0  0.02742     0.71664  0.690
*151
 RCC    .0    .0   5.69468    .0    .0  0.02742     1.1050
*152
 RCC    .0    .0   5.69468    .0    .0  0.02742     1.4351
*153
 RCC    .0    .0   5.69468    .0    .0  0.02742     1.58877
*154
 RCC    .0    .0   5.69468    .0    .0  0.02742     2.0955
*155
 RCC    .0    .0   5.72210    .0    .0  0.02740     0.690
*156
 RCC    .0    .0   5.72210    .0    .0  0.02740     1.1050
*157
 RCC    .0    .0   5.72210    .0    .0  0.02740     1.4351
*158
 RCC    .0    .0   5.72210    .0    .0  0.02740     1.58877
*159
 RCC    .0    .0   5.72210    .0    .0  0.02740     2.0955
*160
 RCC    .0    .0   5.74950    .0    .0  0.081830    0.74334
*161
 RCC    .0    .0   5.74950    .0    .0  0.081830  0.82335
*162
 RCC    .0    .0   5.74950    .0    .0  0.081830  1.1050
*163
 RCC    .0    .0   5.74950    .0    .0  0.081830  1.4351
*164
 RCC    .0    .0   5.74950    .0    .0  0.081830  1.58877
*165
 RCC    .0    .0   5.74950    .0    .0  0.081830  2.0955
*166
 RCC    .0    .0   5.83133    .0    .0  0.070610  0.82335
*167
 RCC    .0    .0   5.83133    .0    .0  0.070610  1.0
*168
 RCC    .0    .0   5.83133    .0    .0  0.070610  1.1050
*169
 RCC    .0    .0   5.83133    .0    .0  0.070610  1.4351
*170
 RCC    .0    .0   5.83133    .0    .0  0.070610  1.58877
*171
 RCC    .0    .0   5.83133    .0    .0  0.070610  2.0955
*172
 RCC    .0    .0   5.90194    .0    .0  0.017560  0.74334
*173
 RCC    .0    .0   5.90194    .0    .0  0.017560  0.82335
```

```
*174
 RCC    .0    .0    5.90194    .0    .0    0.017560   1.0
*175
 RCC    .0    .0    5.90194    .0    .0    0.017560   1.1050
*176
 RCC    .0    .0    5.90194    .0    .0    0.017560   1.4351
*177
 RCC    .0    .0    5.90194    .0    .0    0.017560   1.58877
*178
 RCC    .0    .0    5.90194    .0    .0    0.017560   2.0955
*179
 RCC    .0    .0    5.91950    .0    .0    0.061620   0.74334
*180
 RCC    .0    .0    5.91950    .0    .0    0.061620   0.82335
*181
 RCC    .0    .0    5.91950    .0    .0    0.061620   1.0
*182
 RCC    .0    .0    5.91950    .0    .0    0.061620   1.1050
*183
 RCC    .0    .0    5.91950    .0    .0    0.061620   1.4351
*184
 RCC    .0    .0    5.91950    .0    .0    0.061620   1.58877
*185
 RCC    .0    .0    5.91950    .0    .0    0.061620   2.0955
*186
 RCC    .0    .0    5.98112    .0    .0    0.048870   0.690
*187
 RCC    .0    .0    5.98112    .0    .0    0.048870   1.0
*188
 RCC    .0    .0    5.98112    .0    .0    0.048870   1.1050
*189
 RCC    .0    .0    5.98112    .0    .0    0.048870   1.4351
*190
 RCC    .0    .0    5.98112    .0    .0    0.048870   1.58877
*191
 RCC    .0    .0    5.98112    .0    .0    0.048870   2.0955
*192
 TRC    .0    .0    6.02999    .0    .0    0.08865     0.69     .74334
*193
 RCC    .0    .0    6.02999    .0    .0    0.088650   1.0
*194
 RCC    .0    .0    6.02999    .0    .0    0.088650   1.1050
*195
 RCC    .0    .0    6.02999    .0    .0    0.088650   1.4351
*196
 RCC    .0    .0    6.02999    .0    .0    0.088650   1.58877
*197
 RCC    .0    .0    6.02999    .0    .0    0.088650   2.0955
*198
 RCC    .0    .0    6.11864    .0    .0    0.01270    1.4351
*199
 RCC    .0    .0    6.11864    .0    .0    0.01270    1.58877
*200
 RCC    .0    .0    6.11864    .0    .0    0.01270    2.0955
*201
 RCC    .0    .0    6.13134    .0    .0    0.05088    1.58877
*202
 RCC    .0    .0    6.13134    .0    .0    0.05088    2.0955
*203
 RCC    .0    .0    6.18222    .0    .0    0.05072    1.0
*204
 RCC    .0    .0    6.18222    .0    .0    0.05072    1.58877
*205
 RCC    .0    .0    6.18222    .0    .0    0.05072    2.0955
*206
 RCC    .0    .0    6.23294    .0    .0    2.0        1.0
*207
 RCC    .0    .0    6.23294    .0    .0    2.0        1.58877
*208
 RCC    .0    .0    6.23294    .0    .0    2.0        2.0955
*209
```

```
 RCC     .0     .0     8.23294    .0     .0  0.05080   1.0
*210
 RCC     .0     .0     8.23294    .0     .0  0.05080   1.58877
*211
 RCC     .0     .0     8.23294    .0     .0  0.05080   2.0955
*212
 RCC     .0     .0     8.28374    .0     .0  0.05080   1.58877
*213
 RCC     .0     .0     8.28374    .0     .0  0.05080   2.0955
*214
 RCC     .0     .0     8.33454    .0     .0  0.01270   1.4351
*215
 RCC     .0     .0     8.33454    .0     .0  0.01270   1.58877
*216
 RCC     .0     .0     8.33454    .0     .0  0.01270   2.0955
*217
 TRC     .0     .0     8.34724    .0     .0  0.02743   1.0       0.975
*218
 RCC     .0     .0     8.34724    .0     .0  0.02743   1.4351
*219
 RCC     .0     .0     8.34724    .0     .0  0.02743   1.58877
*220
 RCC     .0     .0     8.34724    .0     .0  0.02743   2.0955
*221
 RCC     .0     .0     8.37467    .0     .0  0.02743   0.975
*222
 RCC     .0     .0     8.37467    .0     .0  0.02743   1.4351
*223
 RCC     .0     .0     8.37467    .0     .0  0.02743   1.58877
*224
 RCC     .0     .0     8.37467    .0     .0  0.02743   2.0955
*225
 RCC     .0     .0     8.40210    .0     .0  0.08179   1.02834
*226
 RCC     .0     .0     8.40210    .0     .0  0.08179   1.10835
*227
 RCC     .0     .0     8.40210    .0     .0  0.08179   1.4351
*228
 RCC     .0     .0     8.40210    .0     .0  0.08179   1.58877
*229
 RCC     .0     .0     8.40210    .0     .0  0.08179   2.0955
*230
 RCC     .0     .0     8.48389    .0     .0  0.07061   1.10835
*231
 RCC     .0     .0     8.48389    .0     .0  0.07061   1.32319
*232
 RCC     .0     .0     8.48389    .0     .0  0.07061   1.4351
*233
 RCC     .0     .0     8.48389    .0     .0  0.07061   1.58877
*234
 RCC     .0     .0     8.48389    .0     .0  0.07061   2.0955
*235
 RCC     .0     .0     8.55450    .0     .0  0.01753   1.02834
*236
 RCC     .0     .0     8.55450    .0     .0  0.01753   1.10835
*237
 RCC     .0     .0     8.55450    .0     .0  0.01753   1.32319
*238
 RCC     .0     .0     8.55450    .0     .0  0.01753   1.4351
*239
 RCC     .0     .0     8.55450    .0     .0  0.01753   1.58877
*240
 RCC     .0     .0     8.55450    .0     .0  0.01753   2.0955
*241
 RCC     .0     .0     8.57203    .0     .0  0.05309   1.02834
*242
 RCC     .0     .0     8.57203    .0     .0  0.05309   1.10835
*243
 RCC     .0     .0     8.57203    .0     .0  0.05309   1.32319
*244
 RCC     .0     .0     8.57203    .0     .0  0.05309   1.4351
```

```
*245
 RCC     .0    .0    8.57203    .0    .0  0.05309    1.58877
*246
 RCC     .0    .0    8.57203    .0    .0  0.05309    2.0955
*247
 RCC     .0    .0    8.62512    .0    .0  0.03987    0.975
*248
 RCC     .0    .0    8.62512    .0    .0  0.03987    1.32319
*249
 RCC     .0    .0    8.62512    .0    .0  0.03987    1.4351
*250
 RCC     .0    .0    8.62512    .0    .0  0.03987    1.58877
*251
 RCC     .0    .0    8.62512    .0    .0  0.03987    2.0955
*252
 TRC     .0    .0    8.66499    .0    .0  0.08865    0.975  1.02834
*253
 RCC     .0    .0    8.66499    .0    .0  0.08865    1.32319
*254
 RCC     .0    .0    8.66499    .0    .0  0.08865    1.4351
*255
 RCC     .0    .0    8.66499    .0    .0  0.08865    1.58877
*256
 RCC     .0    .0    8.66499    .0    .0  0.08865    2.0955
*257
 RCC     .0    .0    8.75364    .0    .0  0.01270    1.4351
*258
 RCC     .0    .0    8.75364    .0    .0  0.01270    1.58877
*259
 RCC     .0    .0    8.75364    .0    .0  0.01270    2.0955
*260
 RCC     .0    .0    8.76634    .0    .0  0.05080    1.58877
*261
 RCC     .0    .0    8.76634    .0    .0  0.05080    2.0955
*262
 RCC     .0    .0    8.81414    .0    .0  0.06604    0.9525
*263
 RCC     .0    .0    8.81414    .0    .0  0.06604    1.58877
*264
 RCC     .0    .0    8.81414    .0    .0  0.06604    2.0955
*265
 RCC     .0    .0    8.88018    .0    .0  0.33180    1.270
*266
 RCC     .0    .0    8.88018    .0    .0  0.33180    1.58877
*267
 RCC     .0    .0    8.88018    .0    .0  0.33180    2.0955
*268
 RCC     .0    .0    9.21198    .0    .0  0.69402    2.0955
*269
 RPP   0.4788   0.6574       -0.5780189      0.5780189  2.61874  4.0665
*270
 RPP   0.4788   0.75057      -0.5780189      0.5780189 2.58058   4.1064
*271
 RPP   0.75057  0.94615      -0.5780189      0.5780189  2.58058  4.1064
*272
 RPP   1.18237  1.36097      -.635    .635    3.380564  4.82854
*273
 RPP   1.18237  1.45415      -.635    .635    3.342464  4.86664
*27
 RPP   1.45415  1.58877      -.635    .635    3.342464  4.86664
*275
 RPP   0.94615  1.0          -.635    .635    6.25348   7.70128
*276
 RPP   0.94615  1.0          -.635    .635    6.23294   6.25348
*277
 RPP   0.94615  1.0          -.635    .635    7.70128   7.73938
*278
 RPP   1.0      1.12475      -.635    .635    6.25348   7.70128
*279
 RPP   1.0      1.12475      -.635    .635    6.23294   6.25348
*280
```

```
  RPP  1.0       1.12475  -.635    .635        6.25348  7.73938
*281
  RPP  1.12475  1.456353 -.635    .635        6.23294  7.73938
*282
  RPP  1.456353 1.58877  -.635    .635        6.23294  7.73938
*283
  RPP  1.584653 1.58877  -.1143   .1143       0.5080   8.88018
*28
  RPP  1.58877 1.8415    -.1143   .1143       0.5080   8.88018
*285
  RCC  .0       .0      10.16    .0       .0     0.3175   2.921
 *286
  RPP  -2.794   4.064   -3.9624  3.9624       10.16     10.4775
*287
  RPP   -2.921 -2.667   -1.191365 1.191365    10.16    10.4775
*288
  RPP   -2.159  4.064   -3.9624   3.9624      9.906    10.16
*289
  RPP  -2.159   4.064   -3.9624   3.9624      0.0      0.254
*290
  RPP   4.064   4.318   -3.9624   3.9624      0.0      10.16
*291
  RPP  -2.794 -2.159    -3.9624   3.9624      0.0      10.16
*292
  RPP  -2.159   4.064    -3.7084   3.9624     0.254     9.906
*293
  RPP  -2.159   4.064    -3.9624  -3.7084     0.254     9.906
*294
  RPP  -2.159   4.064    -3.7084  3.70840     0.254     9.906
*295
  RCC   0.       0.        9.906   0.         0.        0.254            2.0955
*296
  RPP  4.318    5.318   -3.9624   3.9624      0.0     10.4775
*297
  RPP  -3.794   -2.794    -3.9624  3.9624  0.0 10.4775
*298
  RPP  -3.794    5.318    -3.9624  3.9624 -1.0          0.0
*299
  RPP  -3.794    5.318    -3.9624  3.9624  10.4775     11.4775
*300
  RPP  -3.794    5.318    -4.9624 -3.9624  -1.0        11.4775
*301
  RPP  -3.794    5.318     3.9624  4.9624  -1.0        11.4775
*302
  RPP   -1.524   -0.016    3.7084  3.9624   1.508125    8.611575
*303
  RPP   -1.5748 -0.9398 -3.9624 -3.7084  3.889375    6.270625
*304
  RCC  1.8415           0.0      1.6764          0.260   0.  0.  0.08
*305
  RCC  1.8415           0.0      1.6764          0.260   0.  0.  0.08
*306
  RCC  1.8415           0.0      2.21488         0.260  0.  0.  0.08
*307
  RCC  1.8415           0.0      2.46888         0.260  0.  0.  0.08
*308
  RCC  1.8415           0.0      3.01752         0.260  0.  0.  0.08
*309
  RCC  1.8415           0.0      3.27152         0.260  0.  0.  0.08
*310
  RCC  1.8415           0.0      4.21986         0.260  0.  0.  0.08
*311
  RCC  1.8415           0.0      4.46786         0.260  0.  0.  0.08
*312
  RCC  1.8415           0.0      5.88010         0.260  0.  0.  0.08
*313
  RCC  1.8415           0.0      6.13410         0.260  0.  0.  0.08
*314
  RCC  1.8415           0.0      7.01294         0.260  0.  0.  0.08
*315
  RCC  1.8415           0.0      7.26694         0.260  0.  0.  0.08
```

```
*316
  RCC  1.8415          0.0     8.54456         0.260  0. 0. 0.08
*317
  RCC  1.8415          0.0     8.79956         0.260  0. 0. 0.08
*318
  SPH  0.0             0.0     5.0             15.0
  END
*\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

*ZONES
*collimator aperture
  Z01 +1
*Tungsten P17
  Z02 +2 -1
*Tungsten P17
  Z03 +3 -1 -2
*Void
  Z04 +4 -1 -2 -3
* collimator aperture
  Z05 +5
*Tungsten P17
  Z06 +6 -5
*Copper P16
  Z07 +7 -6 -5
*    collimator aperture
  Z08 +8
*Tungsten P17
  Z09 +9 -8
*Copper P16
  Z10 +10 -8 -9 -283
*Copper case cylinder
  Z11 +11 -10 -9 -8 -283 -284
*Copper P16
  Z12 +12  -283
*Copper case cylinder
  Z13 +13 -12 -283 -284
*Plastic  Kel-F  P4
  Z14 +14  -283
*Copper case cylinder
  Z15 +15 -14  -283 -284
*
*D1 Assembly starts here
*
*Void  P9
  Z16 +16
*Phosphor Bronze P9
  Z17 +17 -16
*Plastic  Kel-F  P4
  Z18 +18 -17 -16  -283
*Copper case cylinder
  Z19 +19 -18 -17 -16 -283 -284
*Void
  Z20 +20
*PCB Ring mount
  Z21 +21 -20
  Z22 +22 -21 -20
*Plastic  Kel-F  P8
  Z23 +23 -22 -21 -20
*Plastic  Kel-F  P4
  Z24 +24 -23 -22 -21 -20 -283
*Copper case cylinder
  Z25 +25 -24 -23 -22 -21 -20 -283 -284
*void
  Z26 +26
*PCB Ring mount
  Z27 +27 -26
  Z28 +28 -27 -26
  Z29 +29 -28 -27 -26
*
*
*Plastic  Kel-F  P8
```

```
  Z30 +30 -29 -28 -27 -26
*Plastic  Kel-F  P4
  Z31 +31 -30 -29 -28 -27 -26 -283
*Copper case cylinder
  Z32 +32 -31 -30 -29 -28 -27 -26 -283 -284
*void
  Z33 +33
*Rubber wafer mount
  Z34 +34 -33
*void
  Z35 +35 -34 -33
*PCB Ring mount
  Z36 +36 -35 -34 -33
  Z37 +37 -36 -35 -34 -33
*Plastic  Kel-F  P8
  Z38 +38 -37 -36 -35 -34 -33
*
*Plastic  Kel-F  P4
  Z39 +39 -38 -37 -36 -35 -34 -33 -283
*Copper case cylinder
  Z40 +40 -39 -38 -37 -36 -35 -34 -33 -283 -
284
*Aluminum coating on Si wafer
  Z41 +41
*Oxide ring
  Z42 +42 -41
```

Two-column format used on this and the following five pages to conserve space. Actual file format is single-column.

```
*void
  Z43 +43 -42 -41
*PCB Ring mount
  Z44 +44 -43 -42 -41
  Z45 +45 -44 -43 -42 -41
*Plastic  Kel-F  P8
  Z46 +46 -45 -44 -43 -42 -41
*Plastic  Kel-F  P4
  Z47 +47 -46 -45 -44 -43 -42 -41 -283
*Copper case cylinder
  Z48 +48 -47 -46 -45 -44 -43 -42 -41 -283 -284
*    Silicon wafer
  Z49 +49
*    void
  Z50 +50 -49
*PCB Ring mount
  Z51 +51 -50 -49
*Plastic  Kel-F  P8
  Z52 +52 -51 -50 -49
*Plastic  Kel-F  P4
  Z53 +53 -52 -51 -50 -49 -283
*Copper case cylinder
  Z54 +54 -53 -52 -51 -50 -49  -283 -284
*Aluminum coating on Si wafer
  Z55 +55
*void
  Z56 +56 -55
*PCB Ring mount
  Z57 +57 -56 -55
*Plastic  Kel-F  P8
  Z58 +58 -57 -56 -55
*Plastic  Kel-F  P4
  Z59 +59 -58 -57 -56 -55 -283
*Copper case cylinder
  Z60 +60 -59 -58 -57 -56 -55  -283  -284
*void
  Z61 +61
*PCB Ring mount
  Z62 +62 -61
*Plastic  Kel-F  P8
  Z63 +63 -62 -61
*Plastic  Kel-F  P4
  Z64 +64 -63 -62 -61 -283
*Copper case cylinder
  Z65 +65 -64 -63 -62 -61 -283 -284
*void
  Z66 +66
*
*End of D1 Assembly
*
*Phosphor Bronze P9
  Z67 +67 -66
*Plastic  Kel-F  P4
  Z68 +68 -67 -66 -283
*Copper case cylinder
  Z69 +69 -68 -67 -66  -283 -284
*    Plastic  Kel-F  P6
  Z70 +70 -283
*Copper case cylinder
  Z71 +71 -70  -283 -284
*
*Start of D2 Assembly
*
*Plastic  Kel-F  P4
  Z72   +72   -283
*Copper case cylinder
  Z73   +73  -72 -283 -284
*Void  P9
  Z74   +74
*Phosphor Bronze P9
  Z75   +75  -74
*Plastic  Kel-F  P4
  Z76   +76  -75  -74 -283
*Copper case cylinder
  Z77   +77  -76  -75  -74 -283 -284
*Void
  Z78   +78
*PCB Ring mount
  Z79   +79  -78
  Z80   +80  -79  -78
*Plastic  Kel-F  P8
  Z81   +81  -80  -79  -78
*Plastic  Kel-F  P4
  Z82   +82  -81  -80  -79  -78 -283
*Copper case cylinder
  Z83   +83  -82  -81  -80  -79  -78 -283 -284
*void
  Z84   +84
*PCB Ring mount
  Z85   +85  -84
  Z86   +86  -85  -84
  Z87   +87  -86  -85  -84
*Plastic  Kel-F  P8
  Z88   +88  -87  -86  -85  -84
*Plastic  Kel-F  P4
  Z89   +89  -88  -87  -86  -85  -84  -283
*Copper case cylinder
  Z90   +90  -89  -88  -87  -86  -85  -84  -
283  -284
*void
  Z91   +91
*Rubber wafer mount
  Z92   +92  -91
*void
  Z93   +93  -92  -91
*PCB Ring mount
  Z94   +94  -93  -92  -91
  Z95   +95  -94  -93  -92  -91
*Plastic  Kel-F  P8
  Z96   +96  -95  -94  -93  -92  -91
*Plastic  Kel-F  P4
  Z97   +97  -96  -95  -94  -93  -92  -91 -283
*Copper case cylinder
  Z98   +98  -97  -96  -95  -94  -93  -92  -91
-283 -284
*Aluminum coating on Si wafer
  Z99   +99
*Oxide ring
  Z100  +100  -99
*void
  Z101  +101 -100  -99
*PCB Ring mount
  Z102  +102 -101 -100  -99
  Z103  +103 -102 -101 -100  -99
*Plastic  Kel-F  P8
  Z104  +104 -103 -102 -101 -100  -99
*Plastic  Kel-F  P4
  Z105  +105 -104 -103 -102 -101 -100  -99   -
283
*Copper case cylinder
  Z106  +106 -105 -104 -103 -102 -101 -100  -
99 -283 -284
*    Silicon wafer
  Z107  +107
*    void
  Z108  +108 -107
*PCB Ring mount
  Z109  +109 -108 -107
*Plastic  Kel-F  P8
  Z110  +110 -109 -108 -107
*Plastic  Kel-F  P4
  Z111  +111 -110 -109 -108 -107  -283
```

```
*Copper case cylinder
 Z112  +112 -111 -110 -109 -108 -107  -283 -
284
*Aluminum coating on Si wafer
 Z113  +113
*void
 Z114  +114 -113
*PCB Ring mount
 Z115  +115 -114 -113
*Plastic  Kel-F  P8
 Z116  +116 -115 -114 -113
*Plastic  Kel-F  P4
 Z117  +117 -116 -115 -114 -113 -283
*Copper case cylinder
 Z118  +118 -117 -116 -115 -114 -113 -283 -284
*void
 Z119  +119
*PCB Ring mount
 Z120  +120 -119
*Plastic  Kel-F  P8
 Z121  +121 -120 -119
*Plastic  Kel-F  P4
 Z122  +122 -121 -120 -119  -283
*Copper case cylinder
 Z123  +123 -122 -121 -120 -119  -283 -284
*void
 Z124  +124
*Phosphor Bronze P9
 Z125  +125 -124
*
*    end of D2 assembly
*
*    start of S1, S3 assembly
*
*Plastic  Kel-F  P4
 Z126  +126 -125 -124  -283
*Copper case cylinder
 Z127  +127 -126 -125 -124 -283 -284
*    Plastic  Kel-F  P6
 Z128  +128  -283
*Copper case cylinder
 Z129  +129 -128 -283  -284
*    Plastic  Kel-F  P5
 Z130  +130
* Spectralon P13, P14
 Z131  +131 -130
*    Plastic  Kel-F  P5
 Z132  +132 -131 -130
* Spectralon P13, P14
 Z133  +133 -132 -131 -130  -283
*Copper case cylinder
 Z134  +134 -133 -132 -131 -130 -283   -284
* GSO S1 (will add pin diode later)
 Z135 +135 -269  -270
* Spectralon P13, P14
 Z136  +136 -135  -269 -270 -271
*Plastic Scintillator S3 with pin diode hole
 Z137  +137 -136 -135 -272 -273 -274
* Spectralon P13, P14 with pin diode hole
 Z138  +138 -137 -136 -135 -272 -273 -274 -283
*Copper case cylinder
 Z139  +139 -138 -137 -136 -135 -283 -284
*    Plastic  Kel-F  P5
 Z140  +140
* Spectralon P13, P14
 Z141  +141 -140
*    Plastic  Kel-F  P5
 Z142  +142 -141 -140
* Spectralon P13, P14
 Z143  +143 -142 -141 -140 -283
*Copper case cylinder
 Z144  +144 -143 -142 -141 -140 -283 -284
*    Plastic  Kel-F  P5
 Z145  +145   -283
*Copper case cylinder
 Z146  +146 -145  -283  -284
*
*End of S3, S1 Assembly
*
*Start D3 Assembly
*
* Phosphor Bronze  P9B
 Z147  +147
*Plastic  Kel-F  P5
 Z148  +148 -147  -283
*Copper case cylinder
 Z149  +149 -148 -147  -283 -284
*Void
 Z150  +150
*PCB annulus
 Z151  +151 -150
*Plastic  Kel-F  P7
 Z152  +152 -151 -150
*Plastic  Kel-F  P5,P6
 Z153  +153 -152 -151 -150  -283
*Copper case cylinder
 Z154  +154 -153 -152 -151 -150  -283  -284
*Void
 Z155  +155
*PCB annulus
 Z156  +156 -155
*Plastic  Kel-F  P7
 Z157  +157 -156 -155
*Plastic  Kel-F  P5,P6
 Z158  +158 -157 -156 -155 -283
*Copper case cylinder
 Z159  +159 -158 -157 -156 -155 -283  -284
*Void
 Z160  +160

*Rubber mounting spacer
 Z161  +161 -160
*PMMA
 Z162  +162 -161 -160
*Plastic  Kel-F  P7
 Z163  +163 -162 -161 -160
*Plastic  Kel-F  P5,P6
 Z164  +164 -163 -162 -161 -160  -283
*Copper case cylinder
 Z165  +165 -164 -163 -162 -161 -160 -283 -
284
*Si wafer - electrically active part
 Z166 +166
*Si wafer - electrically inactive part
 Z167 +167 -166
*PMMA
 Z168  +168 -167 -166
*Plastic  Kel-F  P7
 Z169  +169 -168 -167 -166
*Plastic  Kel-F  P5,P6
 Z170  +170 -169 -168 -167 -166 -283
*Copper case cylinder
 Z171  +171 -170 -169 -168 -167 -166  -283 -
284
*Void
 Z172  +172
*Rubber mounting spacer
 Z173  +173 -172
*Void
 Z174  +174 -173 -172
*PMMA
 Z175  +175 -174 -173 -172
```

```
*Plastic  Kel-F  P7
 Z176  +176 -175 -174 -173 -172
*Plastic  Kel-F  P5,P6
 Z177  +177 -176 -175 -174 -173 -172 -283
*Copper case cylinder
 Z178  +178 -177 -176 -175 -174 -173 -172 -283
-284
*Void
 Z179  +179
*Rubber mounting spacer
 Z180  +180 -179
*PMMA
 Z181  +181 -180 -179
*PMMA
 Z182  +182 -181 -180 -179
*Plastic  Kel-F  P7
 Z183  +183 -182 -181 -180 -179
*Plastic  Kel-F  P5,P6
 Z184  +184 -183 -182 -181 -180 -179   -283
*Copper case cylinder
 Z185  +185 -184 -183 -182 -181 -180 -179 -283
-284
*Void
 Z186  +186
*PCB annulus
 Z187  +187 -186
*PMMA
 Z188  +188 -187 -186
*Plastic  Kel-F  P7
 Z189  +189 -188 -187 -186
*Plastic  Kel-F  P5,P6
 Z190  +190 -189 -188 -187 -186  -283
*Copper case cylinder
 Z191  +191 -190 -189 -188 -187 -186 -283 -284
*Void
 Z192  +192
*PCB annulus
 Z193  +193 -192
*PMMA
 Z194  +194 -193 -192
*Plastic  Kel-F  P7
 Z195  +195 -194 -193 -192
*Plastic  Kel-F  P5,P6
 Z196  +196 -195 -194 -193 -192 -283
*Copper case cylinder
 Z197  +197 -196 -195 -194 -193 -192 -283 -284
*Phosphor bronze  P9B
 Z198  +198
*Plastic  Kel-F  P5,P6
 Z199  +199 -198  -283
*Copper case cylinder
 Z200  +200 -199 -198  -283  -284
*Plastic  Kel-F  P6
 Z201  +201  -283
*Copper case cylinder
 Z202  +202 -201 -283 -284
*
*End D3 Assembly
*
*Begin S2 Assembly
*
*Plastic  Kel-F  P4
 Z203  +203
*Spectralon P12
 Z204  +204 -203 -283
*Copper case cylinder
 Z205  +205 -204 -203 -283  -284
*GSO S2
 Z206  +206  -275 -276 -277
*Spectralon P12
 Z207  +207 -206 -275 -276 -277 -278 -279 -
280 -281 -282  -283
*Copper case cylinder
 Z208  +208 -207 -206 -283 -284
*Plastic  Kel-F  P6
 Z209  +209
*Spectralon P12
 Z210  +210 -209  -283
*Copper case cylinder
 Z211  +211 -210 -209 -283 -284
*
*end S2 assembly
*begin D4 assembly
*
*Plastic  Kel-F  P6
 Z212  +212   -283
*Copper case cylinder
 Z213  +213 -212  -283
*Phosphor bronze  P9B
 Z214  +214
*Plastic  Kel-F     P6
 Z215  +215 -214  -283
*Copper case cylinder
 Z216  +216 -215 -214 -283 -284
*Void
 Z217  +217
*PCB annulus
 Z218  +218 -217
*Plastic  Kel-F     P6
 Z219  +219 -218 -217 -283
*Copper case cylinder
 Z220  +220 -219 -218 -217 -283 -284
*Void
 Z221  +221
*PCB annulus
 Z222  +222 -221
*Plastic  Kel-F     P6
 Z223  +223 -222 -221 -283
*Copper case cylinder
 Z224  +224 -223 -222 -221 -283  -284
*Void
 Z225  +225
*Rubber mounting spacer
 Z226  +226 -225
*PMMA
 Z227  +227 -226 -225
*Plastic  Kel-F     P6
 Z228  +228 -227 -226 -225 -283
*Copper case cylinder
 Z229  +229 -228 -227 -226 -225 -283 -284
*Si wafer - electrically active part
 Z230 +230
*Si wafer - electrically inactive part
 Z231 +231 -230
*PMMA
 Z232  +232 -231 -230
*Plastic  Kel-F     P6
 Z233  +233 -232 -231 -230  -283
*Copper case cylinder
 Z234  +234 -233 -232 -231 -230 -283  -284
*Void
 Z235  +235
*Rubber mounting spacer
 Z236  +236 -235
*Void
 Z237  +237 -236 -235
*PMMA
 Z238  +238 -237 -236 -235
*Plastic  Kel-F     P6
 Z239  +239 -238 -237 -236 -235  -283
*Copper case cylinder
```

```
 Z240  +240 -239 -238 -237 -236 -235 -283  -
284
*Void
 Z241  +241
*Rubber mounting spacer
 Z242  +242 -241
*PCB annulus
 Z243  +243 -242 -241
*PMMA
 Z244  +244 -243 -242 -241
*Plastic  Kel-F     P6
 Z245  +245 -244 -243 -242 -241 -283
*Copper case cylinder
 Z246  +246 -245 -244 -243 -242 -241 -283  -
284
*Void
 Z247  +247
*PCB annulus
 Z248  +248 -247
*PMMA
 Z249  +249 -248 -247
*Plastic  Kel-F     P6
 Z250  +250 -249 -248 -247 -283
*Copper case cylinder
 Z251  +251 -250 -249 -248 -247 -283  -284
*Void
 Z252  +252
*PCB annulus
 Z253  +253 -252
*PMMA
 Z254  +254 -253 -252
*Plastic  Kel-F     P6
 Z255  +255 -254 -253 -252  -283
*Copper case cylinder
 Z256  +256 -255 -254 -253 -252  -283 -284
*Phosphor bronze  P9B
 Z257  +257
*PMMA
 Z258  +258 -257   -283
*Copper case cylinder
 Z259  +259 -258 -257 -283 -284
*Plastic  Kel-F     P4
 Z260  +260   -283
*
*end D4 assembly
*Copper case cylinder
 Z261  +261 -260  -283  -284
*Void
 Z262  +262
*Aluminum  p18
 Z263  +263 -262 -283
*Copper case cylinder
 Z264  +264 -263 -262 -283  -284
*Copper base
 Z265  +265
*Aluminum  p18
 Z266  +266 -265
*Copper case cylinder
 Z267  +267 -266 -265
*Copper base
 Z268  +268  OR +295
*Pin Diode Mounted on S1 flat
 Z269  +135  +269
 Z270  +136  +269
*void flat slot for pin diode
*   carved out of S1
 Z271  +135  +270  -269
*carved out of Spectralon
 Z272  +136  +270  -269
*carved out of Spectralon
 Z273  +136  +271
*Pin diode mounted on S3 flat
 Z274  +137  +272
 Z275  +138  +272
*void flat slot for pin diode
*   carved out of S3
 Z276  +273  +137  -272
*carved out of Spectralon
 Z277  +273  +138  -272
*carved out of Spectralon
 Z278  +274  +138
*Void in front of pin diode on S2
 Z279  +206 +276
*Pin diode on S2
 Z280  +206 +275
*Void behind pin diode on S2
 Z281  +206 +277
*Void in front of pin diode on S2
 Z282  +207 +276
*Pin diode on S2
 Z283  +207 +275
*Void behind pin diode on S2
 Z284  +207 +277
*Void in front of pin diode on S2
 Z285  +207 +279
*Pin diode on S2
 Z286  +207 +278
*Void behind pin diode on S2
 Z287  +207 +280
*Void above Pin diode on S2 cut out from
Spectralon
 Z288  +207 +281
 Z289  +207 +282
*Long void slot in copper case to accommodate
pin diode connections
 Z290  +283  +11
 Z291  +283  +13
 Z292  +283  +15
 Z293  +283  +19
 Z294  +283  +25
 Z295  +283  +32
 Z296  +283  +40
 Z297  +283  +48
 Z298  +283  +54
 Z299  +283  +60
 Z300  +283  +65
 Z301  +283  +69
 Z302  +283  +71
 Z303  +283  +73
 Z304  +283  +77
 Z305  +283  +83
 Z306  +283  +90
 Z307  +283  +98
 Z308  +283  +106
 Z309  +283  +112
 Z310  +283  +118
 Z311  +283  +123
 Z312  +283  +127
 Z313  +283  +129
 Z314  +283  +134
 Z315  +283  +139
 Z316  +283  +144
 Z317  +283  +146
 Z318  +283  +149
 Z319  +283  +154
 Z320  +283  +159
 Z321  +283  +165
 Z322  +283  +171
 Z323  +283  +178
 Z324  +283  +185
 Z325  +283  +191
 Z326  +283  +197
```

```
Z327   +283   +200
Z328   +283   +202
Z329   +283   +205
Z330   +283   +208
Z331   +283   +211
Z332   +283   +213
Z333   +283   +216
Z334   +283   +220
Z335   +283   +224
Z336   +283   +229
Z337   +283   +234
Z338   +283   +240
Z339   +283   +246
Z340   +283   +251
Z341   +283   +256
Z342   +283   +259
Z343   +283   +261
Z344   +283   +264
Z345   +284   +11
Z346   +284   +13
Z347   +284   +15
Z348   +284   +19
Z349   +284   +25
Z350   +284   +32
Z351   +284   +40
Z352   +284   +48
Z353   +284   +54
Z354   +284   +60
Z355   +284   +65
Z356   +284   +69
Z357   +284   +71
Z358   +284   +73
Z359   +284   +77
Z360   +284   +83
Z361   +284   +90
Z362   +284   +98
Z363   +284   +106
Z364   +284   +112
Z365   +284   +118
Z366   +284   +123
Z367   +284   +127
Z368   +284   +129
Z369   +284   +134
Z370   +284   +139
Z371   +284   +144
Z372   +284   +146
Z373   +284   +149
Z374   +284   +154
Z375   +284   +159
Z376   +284   +165
Z377   +284   +171
Z378   +284   +178
Z379   +284   +185
Z380   +284   +191
Z381   +284   +197
Z382   +284   +200
Z383   +284   +202
Z384   +284   +205
Z385   +284   +208
Z386   +284   +211
Z387   +284   +213
Z388   +284   +216
Z389   +284   +220
Z390   +284   +224
Z391   +284   +229
Z392   +284   +234
Z393   +284   +240
Z394   +284   +246
Z395   +284   +251
Z396   +284   +256
Z397   +284   +259
Z398   +284   +261
Z399   +284   +264
*Overlap of slot with material zones
  Z400   +12   +283
  Z401   +14   +283
  Z402   +18   +283
  Z403   +24   +283
  Z404   +31   +283
  Z405   +39   +283
  Z406   +47   +283
  Z407   +53   +283
  Z408   +59   +283
  Z409   +64   +283
  Z410   +68   +283
  Z411   +70   +283
  Z412   +72   +283
  Z413   +76    +283
  Z414   +82   +283
  Z415   +89   +283
  Z416   +97   +283
  Z417   +105   +283
  Z418   +111   +283
  Z419   +117   +283
  Z420   +122   +283
  Z421   +126   +283
  Z422   +128   +283
  Z423   +133   +283
  Z424   +138   +283
  Z425   +143   +283
  Z426   +145   +283
  Z427   +148   +283
  Z428   +153   +283
  Z429   +158   +283
  Z430   +164   +283
  Z431   +170   +283
  Z432   +177   +283
  Z433   +184   +283
  Z434   +190   +283
  Z435   +196   +283
  Z436   +199   +283
  Z437   +201   +283
  Z438   +204   +283
  Z439   +207   +283
  Z440   +210   +283
  Z441   +212   +283
  Z442   +215   +283
  Z443   +219   +283
  Z444   +223   +283
  Z445   +228   +283
  Z446   +233   +283
  Z447   +239   +283
  Z448   +245   +283
  Z449   +250   +283
  Z450   +255   +283
  Z451   +258   +283
  Z452   +260   +283
  Z453   +263   +283
  Z454   +10    +283
*Stainless Bulkhead
  Z455   +285
*VOID REGION SURROUNDING bulkhead
  Z456   +286 OR +287 -285
*Aluminum case back plate perp. to z
  Z457   +288 -295
*Aluminum case front plate perp. to z
  Z458   +289 -4
*Aluminum case top plate perp. to x
  Z459   +290
*Aluminum case bottom plate perp. to x
  Z460   +291
*Aluminum case upper side plate perp. to y
  Z461   +292 -302
*Aluminum case lower side plate perp. to y
  Z462   +293 -303
```

```
*Void cavity inside Al case
  Z463  +294 -7 -11 -13 -14 -15
      -19 -25 -32 -40 -48 -54
      -60 -65 -69 -71 -73 -77 -83 -90 -97 -98
      -106 -112 -118 -123 -127 -128 -129 -134 -139 -144 -145
      -146 -149 -154 -159 -165
      -171 -178 -185
      -191 -192 -193 -194 -195
      -196 -197 -198 -199 -200 -201 -202 -203 -204 -205
      -206 -207 -208 -209 -210 -211 -212 -213 -214 -215
      -216 -217 -218 -219 -220 -221 -222 -223 -224 -225
      -226 -227 -228 -229 -230 -231 -232 -233 -234 -235
      -236 -237 -238 -239 -240 -241 -242 -243 -244 -245
      -246 -247 -248 -249 -250 -251 -252 -253 -254 -255
      -256 -257 -258 -259 -260 -261 -262 -263 -264 -265
      -266 -267 -268

*Void region (rectangular) surrounding case
  Z464  +296
  Z465  +297
  Z466  +298
  Z467  +299
  Z468  +300
  Z469  +301
*Large connector hole in upper y plate
  Z470  +302
*Small conector hole in lower y plate
  Z471  +303
*Pinhole #1
  Z472 +304 +15
  Z473 +304 +19
  Z474 +304 +25
  Z475 +304 +32
*Pinhole #2
  Z476 +305 +65
  Z477 +305 +69
*Pinhole #3
  Z478 +306 +83
  Z479 +306 +90
  Z480 +306 +98
  Z481 +306 +106
  Z482 +306 +112
*Pinhole #4
  Z483 +307 +123
  Z484 +307 +127
*Pinhole #5
  Z485 +308 +139
*Pinhole #6
  Z486 +309 +139
*Pinhole #7
  Z487 +310 +139
*Pinhole #8
  Z488 +311 +139
*Pinhole #9
  Z489 +312 +165
  Z490 +312 +171
  Z491 +312 +178
  Z492 +312 +185
*Pinhole #10
  Z493 +313 +197
  Z494 +313 +200
  Z495 +313 +202
  Z496 +313 +205
*Pinhole #11
  Z497 +314 +208
*Pinhole #12
  Z498 +315 +208
*Pinhole #13
Z499 +316 +229
  Z500 +316 +234
  Z501 +316 +240
  Z502 +316 +246
*Pinhole #14
  Z503 +317 +256
  Z504 +317 +261
  Z505 +317 +264
*Escape Sphere
  Z506 +318
      -296 -297 -298 -299 -300 -301
END
```

Eight-column format on the following two pages used to conserve space. Actual file format is single column.

```
*MATERI
AL
*  1
 0
*  2
 3
*  3
 3
*  4
 0
*  5
 0
*  6
 3
*  7
 9
*  8
 0
*  9
 3
* 10
 9
* 11
 9
* 12
 9
* 13
 9
* 14
13
* 15
 9
* 16
 0
* 17
 1
* 18
13
* 19
 9
* 20
 0
* 21
 2
* 22
 2
* 23
13
* 24
13
* 25
 9
* 26
 0
* 27
 2
* 28
 2
* 29
 2
* 30
13
* 31
13
* 32
 9
* 33
 0
* 34
 6
* 35
 0
* 36
 2
* 37
 2
* 38
13
* 39
13
* 40
 9
* 41
 2
* 42
11
* 43
 0
* 44
 2
* 45
 2
* 46
13
* 47
13
* 48
 9
* 49
 8
* 50
 0
* 51
 2
* 52
13
* 53
13
* 54
 9
* 55
 2
* 56
 0
* 57
 2
* 58
13
* 59
13
* 60
 9
* 61
 0
* 62
 2
* 63
13
* 64
13
* 65
 9
* 66
 0
* 67
 1
* 68
13
* 69
 9
* 70
13
* 71
 9
* 72
13
* 73
 9
* 74
 0
* 75
 1
* 76
13
* 77
 9
* 78
 0
* 79
 2
* 80
 2
* 81
13
* 82
13
* 83
 9
* 84
 0
* 85
 2
* 86
 2
* 87
 2
* 88
13
* 89
13
* 90
 9
* 91
 0
* 92
 6
* 93
 0
* 94
 2
* 95
 2
* 96
13
* 97
13
* 98
 9
* 99
 2
*100
11
*101
 0
*102
 2
*103
 2
*104
13
*105
13
*106
 9
*107
 8
*108
 0
*109
 2
*110
13
*111
13
*112
 9
*113
 2
*114
 0
*115
 2
*116
13
*117
13
*118
 9
*119
 0
*120
 2
*121
13
*122
13
*123
 9
*124
 0
*125
 1
*126
13
*127
 9
*128
13
*129
 9
*130
13
*131
12
*132
13
*133
12
*134
 9
*135
 4
*136
12
*137
10
*138
12
*139
 9
*140
13
*141
12
*142
13
*143
12
*144
 9
*145
13
*146
 9
*147
 1
*148
13
*149
 9
*150
 0
*151
 2
*152
13
*153
13
*154
 9
*155
 0
*156
 2
*157
13
*158
13
*159
 9
*160
 0
*161
 6
*162
 7
*163
13
*164
13
*165
 9
*166
 8
*167
 8
*168
 7
*169
13
*170
13
*171
 9
*172
 0
*173
 6
*174
 0
*175
 7
*176
13
*177
13
*178
 9
*179
 0
*180
 6
*181
 7
*182
 7
*183
13
*184
13
*185
 9
*186
 0
*187
 2
*188
 7
*189
13
*190
13
*191
 9
*192
 0
*193
 2
*194
 7
*195
13
*196
13
*197
 9
*198
 1
*199
13
*200
 9
*201
13
*202
 9
*203
13
*204
12
*205
 9
*206
10
*207
12
*208
 9
*209
13
*210
12
*211
 9
*212
13
*213
 9
*214
 1
*215
13
*216
 9
*217
 0
*218
 2
*219
13
*220
 9
*221
 0
*222
 2
*223
13
*224
 9
*225
 0
*226
 6
*227
 7
*228
13
*229
 9
*230
 8
*231
 8
*232
 7
*233
13
*234
 9
*235
 0
*236
 6
*237
 0
*238
 7
*239
13
*240
 9
*241
 0
*242
 6
*243
 2
*244
 7
*245
13
*246
 9
*247
 0
*248
 2
*249
 7
*250
13
*251
 9
*252
 0
*253
 2
*254
 7
*255
13
*256
 9
*257
 1
*258
 7
*259
 9
*260
13
*261
 9
*262
 0
*263
 2
*264
 9
*265
 9
*266
 2
*267
 9
*268
 9
*269
 8
*270
 8
*271
 0
*272
 0
*273
 0
*274
 8
*275
 8
*276
 0
*277
 0
*278
 0
*279
 0
*280
 8
*281
 0
*282
 0
*283
 8
```

*284
0
*285
0
*286
8
*287
0
*288
0
*289
0
*290
0
*291
0
*292
0
*293
0
*294
0
*295
0
*296
0
*297
0
*298
0
*299
0
*300
0
*301
0
*302
0
*303
0
*304
0
*305
0
*306
0
*307
0
*308
0
*309
0
*310
0
*311
0
*312
0
*313
0
*314
0
*315
0
*316
0
*317
0
*318
0
*319
0
*320
0
*321
0
*322
0
*323
0
*324
0
*325
0
*326
0
*327
0
*328
0
*329
0
*330
0
*331
0
*332
0
*333
0
*334
0
*335
0
*336
0
*337
0
*338
0
*339
0
*340
0
*341
0
*342
0
*343
0
*344
0
*345
0
*346
0
*347
0
*348
0
*349
0
*350
0
*351
0
*352
0
*353
0
*354
0
*355
0
*356
0
*357
0
*358
0
*359
0
*360
0
*361
0
*362
0
*363
0
*364
0
*365
0
*366
0
*367
0
*368
0
*369
0
*370
0
*371
0
*372
0
*373
0
*374
0
*375
0
*376
0
*377
0
*378
0
*379
0
*380
0
*381
0
*382
0
*383
0
*384
0
*385
0
*386
0
*387
0
*388
0
*389
0
*390
0
*391
0
*392
0
*393
0
*394
0
*395
0
*396
0
*397
0
*398
0
*399
0
*400
9
*401
13
*402
13
*403
13
*404
13
*405
13
*406
13
*407
13
*408
13
*409
13
*410
13
*411
13
*412
13
*413
13
*414
13
*415
13
*416
13
*417
13
*418
13
*419
13
*420
13
*421
13
*422
13
*423
12
*424
12
*425
12
*426
13
*427
13
*428
13
*429
13
*430
13
*431
13
*432
13
*433
13
*434
12
*435
12
*436
12
*437
13
*438
12
*439
12
*440
12
*441
13
*442
13
*443
13
*444
13
*445
13
*446
13
*447
13
*448
13
*449
13
*450
13
*451
7
*452
13
*453
2
*454
9
*455
5
*456
0
*457
2
*458
2
*459
2
*460
2
*461
2
*462
2
*463
0
*464
0
*465
0
*466
0
*467
0
*468
0
*469
0
*470
0
*471
0
*472
0
*473
0
*474
0
*475
0
*476
0
*477
0
*478
0
*479
0
*480
0
*481
0
*482
0
*483
0
*484
0
*485
0
*486
0
*487
0
*488
0
*489
0
*490
0
*491
0
*492
0
*493
0
*494
0
*495
0
*496
0
*497
0
*498
0
*499
0
*500
0
*501
0
*502
0
*503
0
*504
0
*505
0
*506
0

```
**********************
*********************** SOURCE ********************
ELECTRONS
ENERGY 25.0
POSITION 0.0 0.0  -0.5
    RADIUS 2.1
       DIRECTION  0.0
**********************  OPTIONS  *******************
HISTORIES 100000
```

APPENDIX 2

Annotated ITS-ACCEPT Program Listings Incorporating Disk and Rectangle Source Geometry and Individual History Tracking Options

```
      SUBROUTINE INPUT                                                  INPUT    00007
C ***********************************************************************INPUT    00009
C                                                                       INPUT    00010
C     PROGRAM INPUT IS CALLED BY                                        INPUT    00011
C                                ITS                                    INPUT    00012
C     PROGRAM INPUT CALLS                                               INPUT    00013
C         INTRINSIC FUNCTIONS                                           INPUT    00014
C                               REAL          (TIGER & CYLTRAN)         INPUT    00015
C                               SQRT, ABS     (ACCEPT)                  INPUT    00016
C         EXTERNAL FUNCTIONS                                            INPUT    00017
C                               ALIST,  ELIST,  START,  PREP, KOP,      INPUT    00018
C                               REQALL, GEOMIN, SCRINF, OPOPTS          INPUT    00019
C                               KEYMAP, OPREAD                          INPUT    00020
C                               JOGEN    (ACCEPT)                       INPUT    00021
C                                                                       INPUT    00022
C     ORIGINATION DATE      12 DEC 67.                                  INPUT    00023
C     LAST MODIFIED         17 MAY 91                                   INPUT    00024
C                                                                       INPUT    00025
C     FUNCTION                                                          INPUT    00026
C        THIS PROGRAM IS USED TO READ AND PROCESS USER-SUPPLIED         INPUT    00027
C        CARD INPUT                                                     INPUT    00028
C                                                                       INPUT    00029
C ***********************************************************************INPUT    00030
C *** COMMON BLOCKS CNSTNT, PARAMS, OUT, CALC, XPED, STTS, SCALE, PLTITLINPUT    00031
C                   PAREM, GOMLOC (ACCEPT)                              INPUT    00032
C                   FLUOR         (PCODES)                              INPUT    00033
C                   PLOT          (PLOTS)                               INPUT    00034
C$    LIST(S=0)                                                         INPUT    00035
CDIR$ NOLIST                                                            INPUT    00036
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               CNSTNT   00081
```

No changes in CNSTNT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
      SAVE                                                              CNSTNT   00082
C                                                                       PARAMS   00002
C --------------------------------------------------------------------- PARAMS   00003
C                                                                       PARAMS   00004
C ... I/O UNIT DECLARATIONS AND ARRAY BUFFERS                           PARAMS   00005
      PARAMETER (IIN = 5, IOUT = 6, ITP10 = 10, ITP11 = 11, ITP12 = 12, PARAMS   00006
     $           ITP14 = 14, MAXKEY = 36)
C                                                                       PARAMS   00008
      PARAMETER ( INMT=15,          INEM=8,        INMAX=64,    NSURV=2775, PARAMS   00009
     $           IMTOP=INMAX+1,     IKTOP=89,      IMMAX=33,    INPANG=21,  PARAMS   00010
     $           INRANG=34,  INTANG=INMAX/4+1, INEEL=13,     INPEL=21,    PARAMS   00011
     $           INEPS=9,           INGAS=1000,    INLAN=5000, INPPS=21,    PARAMS   00012
     $           INLAMB=1591,       JAHSUB=51,     IJSPEC=51,   JATPR=698,  PARAMS   00013
     $         JATAN=799, INTAB=30, IMTAX=64)                           STAN     00001
C TMJ: END OF MODIFICATION                                              STAN     00002
C                                                                       PARAMS   00015
      PARAMETER ( IMTOP1 = IMTOP,   INMAX1 = INMAX+1,   INMTP1 = INMT+1, PARAMS   00016
     $           INEEL1 = INEEL,    INGAS1 = INGAS+1,   INLAN1 = INLAN+1,PARAMS   00017
     $           INEPS1 = INEPS,    NSURV1 = NSURV+1,                   PARAMS   00018
     $           INRNG1 = INRANG,   INTNG1 = INTANG  )                  PARAMS   00019
C                                                                       PARAMS   00020
      PARAMETER ( KPTMAX=5000,                      INSTAT=30,          PARAMS   00021
     $           NCHANG=INPANG*INRANG*INTANG,       NJAH1=NCHANG*INMT,  PARAMS   00022
     $           NBDIS  = IKTOP*IMTOP,              NJAH2=NBDIS*INMT,   PARAMS   00023
     $           NGG    = INMAX*IMMAX,              NJAH3=NGG*INMT,     PARAMS   00024
     $           NJAH4 = NSURV*INMT,                NJAH5=JATPR*INMT )  PARAMS   00025
```

Code modification

```
C                                                                    PARAMS    00026
C ... ARRAY DIMENSIONS FOR ZONING AND ESCAPE DISTRIBUTIONS           PARAMS    00027
      PARAMETER (  IKMAX  = 18,   IJMAX  = 50,                       PARAMS    00028
     $             IKPMAX = 18,   IJPMAX = 50,                       PARAMS    00029
     $             INIZON  = 901, INSZON = 900)                      PARAMS    00030
C                                                                    PARAMS    00031
      PARAMETER ( IKMX1  = IKMAX+1,    IKPMX1 = IKPMAX+1,            PARAMS    00032
     $            IJMX1  = IJMAX+1,    IJPMX1 = IJPMAX+1,            PARAMS    00033
     $            IMMAX1 = IMMAX+1 )                                 PARAMS    00034
C                                                                    PARAMS    00035
C ... ARRAY DIMENSIONS FOR PULSE-HEIGHT AND FLUX DISTRIBUTIONS       PARAMS    00036
      PARAMETER ( IJSMAX = 160,         IJFMAX  = 10,                PARAMS    00037
     $            IJSMX1  = IJSMAX+1,   IJFMX1  = IJFMAX+1,          PARAMS    00038
     $            IJFMXP  = 10,         IJFMP1  = IJFMXP+1,          PARAMS    00039
     $            IKFMAX  =  6,         IKFMXP =  6,                 PARAMS    00040
     $            IKFMX1  = IKFMAX+1,   IKFMP1 = IKFMXP+1,           PARAMS    00041
     $            INLF    = 10,         INLFP   = 10)                PARAMS    00042
C                                                                    PARAMS    00043
C ... COMMON AZIMUTHAL PARAMETERS TO FACILITATE COMMON CODING        PARAMS    00044
      PARAMETER (IKMAZ = 1, IKPMAZ = 1)                              PARAMS    00045
      PARAMETER (IKFMAZ = 1, IKFMZP = 1)                             PARAMS    00046
C ______________________________________________                     PARAMS    00048
C ... PARAMETERS SPECIFIC TO ACCEPT AND CYLTRAN                      PARAMS    00049
C ----------------------------------------------                     PARAMS    00050
      PARAMETER (IKMZ1  = IKMAZ+1,    IKPMZ1 = IKPMAZ+1,             PARAMS    00051
     $           IKFMZ1 = IKFMAZ+1,   IKFPZ1 = IKFMZP+1,             PARAMS    00052
     $           INPNTS = 500,                                       PARAMS    00053
     $           NANGS  = 360)                                       PARAMS    00054
C ------------------------------------------------------             PARAMS    00055
C ______________________________                                     PARAMS    00077
C ... ACCEPT SPECIFIC PARAMETERS                                     PARAMS    00078
C ------------------------------                                     PARAMS    00079
      PARAMETER (NESC = 1,      NESC1 = NESC,                        PARAMS    00080
     $ INUMR = 900, NAZ = 5,       ITMA= 18000, IFPD = 6000,         PARAMS    00081
     $ IJTY = 500,  IARB  = 5,     NVIEWS = 5,  NCZONE = 60,         PARAMS    00082
     $ INVALD = 10, INOFND = 10)                                     PARAMS    00083
C ... PARAMETERS FOR AUTOMATIC SUBZONING                             PARAMS    00084
      PARAMETER ( ILSUBZ=4)                                          PARAMS    00085
C -----------------------------------------------------              PARAMS    00086
      PARAMETER (NLAST = 50)                                         PARAMS    00089
      PARAMETER ( INUMK = 3,  INGP = INMT)                           PARAMS    00103
      LOGICAL RRKILL, FLMTEL                                         OUT       00002
```

No changes in OUT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
      COMMON /OUT/                                                   OUT       00003
C                                                                    PLTITL    00002
      CHARACTER*80 TITLE                                             PLTITL    00003
      COMMON /PLTITL/ TITLE                                          PLTITL    00004
C                                                                    PLTITL    00005
C                                                                    CALC      00002
      COMMON /CALC/                                                  CALC      00003
```

No changes in CALC common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C                                                                    CALC      00139
C                                                                    XPED      00002
      COMMON /XPED/                                                  XPED      00003
     1 DETOUR(INMT), RHO(INMT), MT, MTP, MTP0                        XPED      00010
C                                                                    XPED      00012
      LOGICAL DMPFLG, FLMC                                           STTS      00002
      DOUBLE PRECISION IRSAV                                         STTS      00010
      COMMON  /STTS/ IB,  NB,  NSORS, IBT, BOLD, BATCH, KPUTMX, DMPFLG STTS    00017
     $ ,  IHIST,   IRSAV,  KPUT, FLMC                                STTS      00018
C                                                                    STTS      00019
C                                                                    SUBZ      00002
      COMMON /SUBZ/ NSUBZ(INIZON), ZFAC(ILSUBZ)                      SUBZ      00003
     1 , NX(ILSUBZ),     XH(ILSUBZ),     XFAC(ILSUBZ),               SUBZ      00005
     2   NY(ILSUBZ),     YH(ILSUBZ),     YFAC(ILSUBZ),               SUBZ      00006
     3   NZ(ILSUBZ),     ZH(ILSUBZ)                                  SUBZ      00007
     $ , EPS1X(ILSUBZ), EPS1Y(ILSUBZ), EPS1Z(ILSUBZ),                SUBZ      00009
```

```
     $   EPS2X(ILSUBZ), EPS2Y(ILSUBZ), EPS2Z(ILSUBZ)                    SUBZ     00010
C                                                                       PAREM    00002
      CHARACTER*3 OTYPE(10), OBODY                                      PAREM    00003
      LOGICAL FLDBG, FLDBGL                                             PAREM    00004
      COMMON /PAREM/                                                    PAREM    00008
     $ XB(3),     WT(3),     RIN,    ROUT,    PINF,    DIST,    IR,     PAREM    00009
     $ FLDBG,     IRPRIM,    ICALL,  LSURF,   NBO,     LRI,     LRO,    PAREM    00013
     $ KLOOP,     LOOP,      ITYPE,  FLDBGL                             PAREM    00014
      COMMON /PAREMO/  OTYPE                                            PAREM    00015
C                                                                       PAREM    00016
C                                                                       GOMLOC   00002
      COMMON /GOMLOC/                                                   GOMLOC   00003
     $   LDATA,  LTMA,  LFPD,  NUMR,  NUMB, LDATP1, LTMAM1,             GOMLOC   00004
     $   NVALD,  NOFND                                                  GOMLOC   00005
C                                                                       GOMLOC   00006
C                                                                       PLOT     00002
      COMMON /PLOT/ NPLOTS, PHIPLT(NVIEWS), THEPLT(NVIEWS),              PLOT     00003
     $   XMNPLT(NVIEWS), XMXPLT(NVIEWS), YMNPLT(NVIEWS), YMXPLT(NVIEWS), PLOT     00004
     $   XMN, XMX, YMN, YMX, KPLT                                       PLOT     00005
C$    LIST(S=1)                                                         INPUT    00051
CDIR$ LIST                                                              INPUT    00052
      COMMON  /SCALE/  BNUM, XNUM                                       SCALE    00002
        COMMON /EXTSORC/ IRECTS,IDISKS,XLOWS,XHIGHS,YLOWS,YHIGHS,ZLOWS,           New code
     $ ZHIGHS,XCENT,YCENT,ZCENT,XCIR,YCIR,ZCIR,KPERPYZ,KPREPXZ,KPERPXY
C                                                                       SCALE    00003
C                                                                       INPUT    00057
        COMMON /HITS/EDPR(10),EDNK(10),EDSC(10),EDTL(10),LHCL(10),NINDV           New code

      CHARACTER*80    KARD                                              INPUT    00058
      COMMON / IOPACK   / KARD                                          INPUT    00059
      LOGICAL EOFLAG, FLKEY(MAXKEY), FLDUP, FLNEWD                      INPUT    00060
C                                                                       INPUT    00061
      WRITE(IOUT,'(''1*****************************''/                   INPUT    00062
     $              '' *      BEGIN READING INPUT     *''/               INPUT    00063
     $              '' *****************************'')')               INPUT    00064
      IF (IRUN .NE. 1) THEN                                             INPUT    00065
C                                                                       INPUT    00066
C       ****************************************************************INPUT    00067
C       * CONVERT UNITS FROM CM BACK TO GM/CM**2 FOR MULTIPLE PROBLEMS *INPUT    00068
C       ****************************************************************INPUT    00069
C                                                                       INPUT    00070
        DO 50 L=1,NMT                                                   INPUT    00071
          RHOL = RHO(L)                                                 INPUT    00073
          DO 10 N=1,NMAX1                                               INPUT    00074
            RANGE(N,L) = RANGE(N,L)*RHOL                                INPUT    00075
            DRG(N,L)   = DRG(N,L)   *RHOL                               INPUT    00076
            PXRAY(N,L) = PXRAY(N,L)/RHOL                                INPUT    00077
            PBREM(N,L) = PBREM(N,L)/RHOL                                INPUT    00078
   10     CONTINUE                                                      INPUT    00079
          DO 20 N=1,NMAX                                                INPUT    00080
            DRGS(N,L)   = DRGS(N,L) *RHOL                               INPUT    00081
            COSAV(N,L) = COSAV(N,L)/RHOL                                INPUT    00082
   20     CONTINUE                                                      INPUT    00083
          DO 30 J=1, NGMAX                                              INPUT    00084
            AT(J,L) = AT(J,L)/RHOL                                      INPUT    00085
   30     CONTINUE                                                      INPUT    00086
   50   CONTINUE                                                        INPUT    00087
      END IF                                                            INPUT    00088
C     **********************************                                INPUT    00133
C     *  SET DEFAULT INPUT PARAMETERS   *                                INPUT    00134
C     **********************************                                INPUT    00135
      FLSTRG = .TRUE.                                                   INPUT    00136
      FLNOK  = .TRUE.                                                   INPUT    00137
      FLNEL  = .FALSE.                                                  INPUT    00138
      FLBAD  = .TRUE.                                                   INPUT    00139
      FLGSEC = .TRUE.                                                   INPUT    00140
      FLNXEV = .FALSE.                                                  INPUT    00141
      FLBSC  = .FALSE.                                                  INPUT    00142
      FLCOH  = .TRUE.                                                   INPUT    00143
      FLSKN  = .TRUE.                                                   INPUT    00144
      FLDBG  = .FALSE.                                                  INPUT    00146
```

```
      FLDBGL = .FALSE.                                                   INPUT    00147
      RLAN   = C5EM1                                                     INPUT    00149
      BNUM   = CZERO                                                     INPUT    00150
      XNUM   = CZERO                                                     INPUT    00151
      DLIM   = CZERO                                                     INPUT    00152
      NPRTCL = 1                                                         INPUT    00153
C
      IRECTS = 0
        IDISKS = 0
        KPERPYZ = 0
        KPERPXZ = 0
        KPERPXY = 0                    New code
C
      NINDV=0
        DO 599 J=1,10
599   LHCL(J)=0
C                                                                        INPUT    00154
      TITLE  = ' '                                                       INPUT    00155
      NPRT   = 12                                                        INPUT    00156
      IECHO  = 0                                                         INPUT    00157
      NB     = 10                                                        INPUT    00158
      IMAX   = 1000                                                      INPUT    00159
      IBT    = 0                                                         INPUT    00160
      MBSC   = 1                                                         INPUT    00161
      BOLD   = CZERO                                                     INPUT    00162
      IMXOLD = 0                                                         INPUT    00163
      INRAN  = CZERO                                                     INPUT    00168
      BASE   = CTWO                                                      INPUT    00170
      XNCYC  = CEIGHT                                                    INPUT    00171
      TMFAC  = BASE**(-1.0/XNCYC)                                        INPUT    00172
      DMPFLG = .FALSE.                                                   INPUT    00173
C                                                                        INPUT    00174
C ... INITIALIZE LOGICALS FOR IDENTIFYING MATERIALS (NON-P CODES) OR     INPUT    00175
C     ELEMENTS (P CODES) THAT ARE PRESENT IN A GIVEN PROBLEM - USED      INPUT    00176
C     FOR IDENTIFYING RELEVANT LINE RADIATION.                           INPUT    00177
      NGP = NMT                                                          INPUT    00182
      DO 60 J=1,NGP                                                      INPUT    00184
   60   FLMTEL(J) = .FALSE.                                              INPUT    00185
C                                                                        INPUT    00186
      NPLOTS  = 0                                                        INPUT    00188
C    -----------------                                                   INPUT    00193
C ... SOURCE VARIABLES                                                   INPUT    00194
C    -----------------                                                   INPUT    00195
      FLESRC = .TRUE.                                                    INPUT    00196
      JSPEC  = 0                                                         INPUT    00197
      FLSPEC = .FALSE.                                                   INPUT    00198
      TIN    = CONE                                                      INPUT    00199
      TPCUT  = C1EM2                                                     INPUT    00200
      TCUT   = CZERO                                                     INPUT    00201
      TSAVE  = CZERO                                                     INPUT    00202
      ICTH   = 1                                                         INPUT    00203
      CTSR   = CZERO                                                     INPUT    00204
      CTHIN  = C90                                                       INPUT    00205
      ZSR    = CZERO                                                     INPUT    00210
      XSR    = CZERO                                                     INPUT    00213
      YSR    = CZERO                                                     INPUT    00214
      CPSR   = CZERO                                                     INPUT    00215
      SORCIN = CZERO                                                     INPUT    00216
C    --------------------------                                          INPUT    00218
C ... ELECTRON ESCAPE VARIABLES                                          INPUT    00219
C    --------------------------                                          INPUT    00220
      JMAX  = 10                                                         INPUT    00221
      FLESC = .FALSE.                                                    INPUT    00222
      ITMK  = 1                                                          INPUT    00223
      IAMK  = 1                                                          INPUT    00224
      KMAX  = 18                                                         INPUT    00225
      KMAZ  = 1                                                          INPUT    00226
      IAMKZ = 1                                                          INPUT    00228
C    ------------------------                                            INPUT    00230
C ... PHOTON ESCAPE VARIABLES                                            INPUT    00231
C    ------------------------                                            INPUT    00232
```

```
      JPMAX  = 10                                                       INPUT   00233
      FLESCP = .FALSE.                                                  INPUT   00234
      IPMK   = 1                                                        INPUT   00235
      IBMK   = 1                                                        INPUT   00236
      KPMAX  = 18                                                       INPUT   00237
      KPMAZ  = 1                                                        INPUT   00238
      IBMKZ  = 1                                                        INPUT   00240
C     ------------------------                                          INPUT   00242
C ... ELECTRON FLUX VARIABLES                                           INPUT   00243
C     ------------------------                                          INPUT   00244
      FLFLUX = .FALSE.                                                  INPUT   00245
      JFMAX  = 10                                                       INPUT   00246
      KFMAX  = 6                                                        INPUT   00247
      KFMAZ  = 1                                                        INPUT   00248
      IFAMKZ = 1                                                        INPUT   00250
      IFMK   = 1                                                        INPUT   00252
      IFAMK  = 1                                                        INPUT   00253
C     ----------------------                                            INPUT   00254
C ... PHOTON FLUX VARIABLES                                             INPUT   00255
C     ----------------------                                            INPUT   00256
      FLFLXP = .FALSE.                                                  INPUT   00257
      JFMAXP = 10                                                       INPUT   00258
      KFMAXP = 6                                                        INPUT   00259
      KFMAZP = 1                                                        INPUT   00260
      IFBMKZ = 1                                                        INPUT   00262
      IFMKP  = 1                                                        INPUT   00264
      IFBMK  = 1                                                        INPUT   00265
C     -----------------------------------                               INPUT   00266
C ... PULSE HEIGHT DISTRIBUTION VARIABLES                               INPUT   00267
C     -----------------------------------                               INPUT   00268
      FLPHD  = .FALSE.                                                  INPUT   00269
      JSMAX  = 12                                                       INPUT   00270
      IPHMK  = 1                                                        INPUT   00271
C                                                                       INPUT   00272
C     ***********************************************                   INPUT   00273
C     *            BEGIN READING INPUT                  *               INPUT   00274
C     *  ZERO-LEVEL KEYWORDS IN ALPHABETICAL ORDER      *               INPUT   00275
C     ***********************************************                   INPUT   00276
C                                                                       INPUT   00280
C ... SET ERROR TRAP FLAG TO ZERO                                       INPUT   00281
      IERTRP = 0                                                        INPUT   00282
      NUMCRD = 0                                                        INPUT   00283
      FLNEWD = .FALSE.                                                  INPUT   00284
      FLDUP  = .FALSE.                                                  INPUT   00285
      DO 65 IKEY=1,MAXKEY                                               INPUT   00286
   65   FLKEY(IKEY) = .FALSE.                                           INPUT   00287
C                                                                       INPUT   00288
C ... READ THE NEXT CARD IN THE INPUT FILE                              INPUT   00289
C                                                                       INPUT   00290
C                                                                       INPUT   00291
   70 CALL OPREAD(1,IECHO,EOFLAG)                                       INPUT   00292
C     ---------------------------                                       INPUT   00293
C                                                                       INPUT   00294
C ... NOTE, COMMENT CARDS DENOTED BY * IN COLUMN 1, SKIPPED INTERNALLY  INPUT   00295
C                                                                       INPUT   00296
      IF (.NOT. EOFLAG) THEN                                            INPUT   00297
        NUMCRD = NUMCRD + 1                                             INPUT   00298
C                                                                       INPUT   00299
   80   IF (KOP('BATCHES') .GE. 1) THEN                                 INPUT   00300
C         ---------                                                     INPUT   00301
C ...      BATCHES                                                      INPUT   00302
C         ---------                                                     INPUT   00303
C          Check if primary keyword has been used                       INPUT   00304
C                                                                       INPUT   00305
           IKEY = 1                                                     INPUT   00306
C                                                                       INPUT   00307
           IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)                     INPUT   00308
C                           ------------------------                    INPUT   00309
           FLKEY(IKEY) = .TRUE.                                         INPUT   00310
C                                                                       INPUT   00311
           NB = PARM(1)                                                 INPUT   00312
```

```
C
         ELSE IF (KOP('CUTOFFS') .GE. 0) THEN
C          ---------
C ...       CUTOFFS
C          ---------
            IKEY = 2
C
            IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                            ------------------------
            FLKEY(IKEY) = .TRUE.
C
            KARG = KOP('CUTOFFS')
            IF (KARG .GE. 1) THEN
              TCUT = PARM(1)
            END IF
            IF (KARG .GE. 2) THEN
              TPCUT = PARM(2)
            END IF
C
         ELSE IF (KOP('DETAIL-IONIZE') .GE. 0) THEN
C
C ...       DETAIL-IONIZATION
C          -------------------
            IKEY = 33
C
            IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                            ------------------------
            FLKEY(IKEY) = .TRUE.
C
            NPRTCL = 2
C

C
         ELSE IF (KOP('RECTANGLE-SOURCE').GE.0)THEN
C
C
C           RECTANGULAR PLANE SOURCE
C          ------------

            IKEY = 34
C
C
            IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                            ------------------------
            FLKEY(IKEY) = .TRUE.
C

            KARG = KOP('RECTANGLE-SOURCE')
        IF(KARG.LT.6)THEN
                 WRITE(IOUT,68)

68     FORMAT(1X,'>>>>>')
                 WRITE(IOUT,51)
                     WRITE(IOUT,68)

51     FORMAT(1X,' USER MUST ENTER 6 NUMBERS (XLOW,XHIGH,YLOW,YHIGH,ZLOW,
      $ZHIGH) TO DEFINE SOURCE LOWER AND UPPER COORDINATE LIMITS OF SOURC
      $E RECTANGLE')
             CALL ABORTX('INPUT')
        ELSE
             IRECTS = 1
        XLOWS = PARM(1)
        XHIGHS = PARM(2)
        YLOWS = PARM(3)
        YHIGHS = PARM(4)
        ZLOWS = PARM(5)
        ZHIGHS = PARM(6)
        END IF
```

New code

```
C
      ELSE IF (KOP('CIRCLE-SOURCE').GE.0)THEN
C          ___________
C           CIRCLE PLANE SOURCE
C          ------------
           IKEY = 35
C
C
           IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                           -----------------------
           FLKEY(IKEY) = .TRUE.
C

           KARG = KOP('CIRCLE-SOURCE')
        IF(KARG.LT.6)THEN
               WRITE(IOUT,68)

               WRITE(IOUT,52)

               WRITE(IOUT,68)

52    FORMAT(1X,' USER MUST ENTER 6 NUMBERS - COORDINATES OF CIRCLE CENT
     $ER (XO,YO,ZO),AND COORDINATES A POINT ON CIRCUMFERENCE'/1X,'(XC,YC
     $,ZC) TO DEFINE POSITION AND ORIENTATION OF SOURCE CIRCLE')
C

             CALL ABORTX('INPUT')
        ELSE
             IDISKS = 1
        XCENT = PARM(1)
        YCENT = PARM(2)
        ZCENT = PARM(3)
        XCIR = PARM(4)
        YCIR = PARM(5)
        ZCIR = PARM(6)
        CALL OPREAD(1,IECHO,EOFLAG)
        IF(KOP('RADIUS').GE.1) THEN
        SORCIN = PARM(1)
        ELSE
        GO TO 80
        END IF

C

      END IF
C
C
      ELSE IF (KOP('INDIVIDUAL-HISTS').GE.0) THEN
C       ________________________________________
C         RECORD SINGLE HISTORY ENERGY DEPOSITIONS
C        -----------------------------------------
C
          IKEY = 36
C
          IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C
          FLKEY(IKEY) = .TRUE.
C
          KARG    = KOP('INDIVIDUAL-HISTS')
            IF(KARG.LT.1.OR. KARG.GT.10)THEN
                 WRITE(IOUT,68)
                      WRITE(IOUT,688)
                      WRITE(IOUT,68)
688   FORMAT(1X,'USER MUST ENTER NO FEWER THAN 1 AND NO MORE THAN 10 CEL
     $L NUMBERS IN WHICH THE ENERGY DEPOSITION'/1X,'FOR INDIVIDUAL ELECT
     $RON HISTORIES ARE TO BE RECORDED.')
C
```

New code

```
          CALL ABORTX('INPUT')
        ELSE
          DO 689 KRRG=1,KARG
689      LHCL(KRRG)=PARM(KRRG)
         NINDV=KARG
        WRITE(IOUT,587)
        WRITE(IOUT,588)(LHCL(KRRG),KRRG=1,NINDV)
588    FORMAT(1X,'ENERGY DEPOSITION FOR INDIVIDUAL HISTORIES WILL BE RECO
     $RDED ON FILE "EDSHOW.TXT" FOR CELL NOS.'/5X,10I5)
        WRITE(IOUT,587)
587    FORMAT(/1X,'**********************************************************
     $**********************************************************'
     $/1X,'***********************************************************
     $**********************************************************')
       END IF
```

New code

```
C

C
       ELSE IF (KOP('DIRECTION') .GE. 0) THEN
C
C ...     DIRECTION
C         -----------
          IKEY = 3
C
          IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                          -------------------------
          FLKEY(IKEY) = .TRUE.
C
```

Remaining portion of subroutine INPUT (omitted here for brevity) is identical to original ACCEPT [1] code

•
•
•
•
•
•
•
•
•

```
       END
```

```
      SUBROUTINE HIST                                                   HIST     00007
C ***********************************************************************HIST     00009
C                                                                       HIST     00010
C     SUBROUTINE HIST IS CALLED BY                                      HIST     00011
C                                  ITS                                  HIST     00012
C     SUBROUTINE HIST CALLS                                             HIST     00013
C         INTRINSIC FUNCTIONS                                           HIST     00014
C                                    SQRT, RANF                         HIST     00015
C                                    REAL                (CYLTRAN)      HIST     00016
C         EXTERNAL FUNCTIONS                                            HIST     00017
C                                    CLASS, ECROS, EHIST, TIMER, PHIST  HIST     00018
C                                    RANINT, RANSAV                     HIST     00019
C                                    ZONE                      (CYLTRAN) HIST    00020
C                                    FOLD,  ZONEA              (ACCEPT)  HIST    00021
C                                    PLTDAT                    (M-CODES) HIST    00022
C                                                                       HIST     00023
C     ORIGINATION DATE      16 JAN 68.                                  HIST     00024
C     LAST MODIFIED         30 MAY 91                                   HIST     00025
C                                                                       HIST     00026
C     FUNCTION                                                          HIST     00027
C        THIS PROGRAM SAMPLES PHASE SPACE PARAMETERS FOR                HIST     00028
C        SOURCE PARTICLES.  SUBSEQUENTLY CALLS EITHER EHIST OR          HIST     00029
C        PHIST.  RETRIEVES "BANKED" ELECTRONS AND CALLS EHIST.          HIST     00030
C        TALLIES PULSE HEIGHT DISTRIBUTION.                             HIST     00031
C                                                                       HIST     00032
C ***********************************************************************HIST     00033
C *** COMMON BLOCKS CNSTNT, PARAMS, OUT, CALC, XPED, STOR, STTS,        HIST     00034
C                     (PAREM)-ACCEPT                                    HIST     00035
C$    LIST(S=0)                                                         HIST     00036
CDIR$ NOLIST                                                            HIST     00037
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               CNSTNT   00081
```

No changes in CNSTNT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
      PARAMETER (CCOHLM=57.031547D0,    CCOHMX=80.654788D0)             CNSTNT   00140
C                                                                       PARAMS   00002
C ---------------------------------------------------------------------- PARAMS  00003
C                                                                       PARAMS   00004
C ... I/O UNIT DECLARATIONS AND ARRAY BUFFERS                           PARAMS   00005
      PARAMETER (IIN = 5, IOUT = 6, ITP10 = 10, ITP11 = 11, ITP12 = 12, PARAMS   00006
```

PARAMS common block identical to that shown in subroutine INPUT

```
        COMMON /EXTSORC/ IRECTS,IDISKS,XLOWS,XHIGHS,YLOWS,YHIGHS,ZLOWS,
     $  ZHIGHS,XCENT,YCENT,ZCENT,XCIR,YCIR,ZCIR,KPERPYZ,KPREPXZ,KPERPXY
      LOGICAL RRKILL, FLMTEL                                            OUT      00002
      COMMON /OUT/                                                      OUT      00003
```

New Code

No changes in OUT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C                                                                       CALC     00002
      COMMON /CALC/                                                     CALC     00003
```

No changes in CALC common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C                                                                       XPED     00002
      COMMON /XPED/                                                     XPED     00003
     1 DETOUR(INMT), RHO(INMT), MT, MTP, MTP0                           XPED     00010
C                                                                       XPED     00012
      LOGICAL DMPFLG, FLMC                                              STTS     00002
      DOUBLE PRECISION IRSAV                                            STTS     00010
      COMMON  /STTS/  IB,  NB,  NSORS, IBT, BOLD, BATCH, KPUTMX, DMPFLG STTS     00017
     $ ,  IHIST,   IRSAV,  KPUT, FLMC                                   STTS     00018
C                                                                       STTS     00019
C                                                                       PAREM    00002
      CHARACTER*3 OTYPE(10), OBODY                                      PAREM    00003
      LOGICAL FLDBG, FLDBGL                                             PAREM    00004
      COMMON /PAREM/                                                    PAREM    00008
     $ XB(3),   WT(3),   RIN,   ROUT,   PINF,   DIST,   IR,             PAREM    00009
```

```
     $ FLDBG,   IRPRIM,  ICALL, LSURF,  NBO,     LRI,     LRO,          PAREM    00013
     $ KLOOP,   LOOP,    ITYPE, FLDBGL                                  PAREM    00014
      COMMON /PAREMO/  OTYPE                                            PAREM    00015
C                                                                       PAREM    00016
         COMMON /HITS/EDPR(10),EDNK(10),EDSC(10),EDTL(10),LHCL(10),NINDV
```

New Code

```
C$    LIST(S=1)                                                         HIST     00047
CDIR$ LIST                                                              HIST     00048
      COMMON /STOR/                                                     STOR     00002
     1 CTHS(NLAST),  TS(NLAST),  WS(NLAST),  ZS(NLAST),  IPRS(NLAST),   STOR     00003
     2 LBS(NLAST),   NTS(NLAST)                                         STOR     00004
     $ ,XS(NLAST),   YS(NLAST),  STHS(NLAST),                           STOR     00006
     3 CPHS(NLAST),  SPHS(NLAST)                                        STOR     00007
     4 ,LBCS(NLAST)                                                     STOR     00009
C                                                                       HIST     00050
      EXTERNAL RAN                                                      RANNUM   00003
C                                                                       HIST     00089
      CIMAX = IMAX                                                      HIST     00090
      IF (FLSPEC) THEN                                                  HIST     00091
        TAV = CZERO                                                     HIST     00092
      ELSE                                                              HIST     00093
        TAV = CIMAX*TIN                                                 HIST     00094
      END IF                                                            HIST     00095
C    ___________________                                                HIST     00096
      CALL RANINT(IRA)                                                  HIST     00097
C    -------------------                                                HIST     00098
      IF (IB .EQ. 1) INRAN = IRA                                        HIST     00101
      DO 130 I = 1, IMAX                                                HIST     00103
           DO 1301 JJJ=1,10
               EDPR(JJJ)=0.
               EDNK(JJJ)=0.
               EDSC(JJJ)=0.
1301    EDTL(JJJ)=0.
```

New Code

```
        IHIST = I                                                       HIST     00104
      MODTMJ = MIN(100,IMAX)                                            LAHEY    00017
      IF(I.EQ.MODTMJ*(I/MODTMJ)) THEN                                   LAHEY    00018
      CALL TOTTIM(XTMJ)                                                 LAHEY    00019
      WRITE(*,'(/'' HISTORY'',I8,'', ELAPSED MINUTES'',F10.2)')         LAHEY    00020
     1I,XTMJ/60.                                                        LAHEY    00021
      ENDIF                                                             LAHEY    00022
        W      = CONE                                                   HIST     00105
        CWCF   = W                                                      HIST     00106
        LAST   = 0                                                      HIST     00107
C       ____________________                                            HIST     00108
        CALL RANSAV(IRSAV)                                              HIST     00109
C       --------------------                                            HIST     00110
C                                                                       HIST     00111
C    __________________                                                 HIST     00112
C ... SOURCE ENERGY                                                     HIST     00113
C ------------------                                                    HIST     00114
        IF (FLSPEC) THEN                                                HIST     00115
          RA = RAN(IRAN)                                                HIST     00116
          DO 14 JHIST = 2,JSPEC                                         HIST     00117
             IF ( RA. GT. SPECIN(JHIST) ) GO TO 16                      HIST     00118
   14     CONTINUE                                                      HIST     00119
   16     T = ESP(JHIST-1) + ( RA -SPECIN(JHIST-1) )*( ESP(JHIST)       HIST     00120
     $      - ESP(JHIST-1) )/( SPECIN(JHIST) - SPECIN(JHIST-1) )        HIST     00121
          TAV = TAV + T                                                 HIST     00122
          IF ( (FLESRC .AND. (T .GT. TCUT )) .OR.                       HIST     00123
     $          (.NOT. FLESRC .AND. (T .GT. TPCUT)) ) THEN              HIST     00124
              GO TO 20                                                  HIST     00125
          ELSE                                                          HIST     00126
              NTREJ = NTREJ + 1                                         HIST     00127
              TREJ  = TREJ + W*T                                        HIST     00128
              GO TO 1299                                                HIST     00129
          END IF                                                        HIST     00130
        END IF                                                          HIST     00131
        T  = TIN                                                        HIST     00132
   20   NT = NTFST                                                      HIST     00133
C       __________________                                              HIST     00134
        CALL CLASS (T,NT)                                               HIST     00135
C       ------------------                                              HIST     00136
```

```
C ___________________                                        HIST   00137
C ... SOURCE DIRECTION                                       HIST   00138
C ---------------------                                      HIST   00139
        IF (ICTH .EQ. 2) THEN                                HIST   00140
          RA  = RAN(IRAN)                                    HIST   00141
          COM = CTHIN+ RA*(CONE-CTHIN)                       HIST   00142
        ELSE IF (ICTH .EQ. 3) THEN                           HIST   00143
          RA  = RAN(IRAN)                                    HIST   00144
          COM = SQRT(CTHIN+RA*(CONE-CTHIN))                  HIST   00145
        ELSE IF (ICTH .EQ. 1) THEN                           HIST   00146
          CTH(1) = CTSR                                      HIST   00147
          STH(1) = STSR                                      HIST   00149
          CPH(1) = CPSR                                      HIST   00150
          SPH(1) = SPSR                                      HIST   00151
          GO TO 69                                           HIST   00153
        END IF                                               HIST   00154
C                                                            HIST   00155
        IF (CTSR .EQ. CONE) THEN                             HIST   00156
          CTH(1) = COM                                       HIST   00157
          STH(1) = SQRT(CONE-COM*COM)                        HIST   00159
          RA     = RAN(IRAN)                                 HIST   00160
          JAZ    = RA*C360                                   HIST   00161
          CPH(1) = CCH(JAZ+1)                                HIST   00162
          SPH(1) = SCH(JAZ+1)                                HIST   00163
        ELSE                                                 HIST   00165
C        ______________________________________________________HIST   00172
          CALL FOLD(CTSR,STSR,CPSR,SPSR,COM,CTH(1),STH(1),CPH(1),SPH(1))HIST   00173
C        ------------------------------------------------------HIST   00174
        END IF                                               HIST   00176
C ___________________                                        HIST   00177
C ... SOURCE POSITION                                        HIST   00178
C --------------------                                       HIST   00179
 69     IF (SORCIN .NE. CZERO) THEN                          HIST   00198
          RA   = RAN(IRAN)                                   HIST   00199
          R    = SQRT(RA)*SORCIN                             HIST   00200
          RA   = RAN(IRAN)                                   HIST   00201
          JAZ  = RA*C360                                     HIST   00202
          SCHR = SCH(JAZ+1)*R                                HIST   00203
          CCHR = CCH(JAZ+1)*R                                HIST   00204
            IF (IDISKS .EQ. 0) THEN
                  X   = XSR + CCHR*W1X+SCHR*W2X
                  Y   = YSR+CCHR*W1Y+SCHR*W2Y
                  Z   = ZSR+CCHR*W1Z+SCHR*W2Z
              ELSE
                 IF(KPERPXY.EQ.1)THEN
                   X = XCENT + CCHR
                     Y = YCENT + SCHR
                     Z = ZCENT
                   END IF
                   IF (KPERPXZ.EQ.1)THEN
                     X = XCENT + CCHR
                     Y = YCENT
                     Z = ZCENT + SCHR
                   END IF
                   IF(KPERPYZ.EQ.1)THEN

                X = XCENT
                     Y = YCENT + CCHR

                Z = ZCENT + SCHR

              END IF

            END IF

        ELSE
            IF (IRECTS .EQ. 0) THEN                          HIST   00208
          X = XSR                                            HIST   00209
          Y = YSR                                            HIST   00210
          Z = ZSR                                            HIST   00211
```

New Code

New Code

```
ELSE

          RRAA1 = RAN(IRAN)
          RRAA2 = RAN(IRAN)

          IF (KPERPXY .EQ. 1) THEN
            X = XLOWS + RRAA1*(XHIGHS-XLOWS)
            Y = YLOWS + RRAA2*(YHIGHS-YLOWS)
            Z = ZLOWS

          END IF

          IF (KPERPXZ. EQ. 1) THEN
            X = XLOWS + RRAA1*(XHIGHS-XLOWS)
            Y = YLOWS

            Z = ZLOWS + RRAA2*(ZHIGHS-ZLOWS)
          END IF

          IF (KPERPYZ .EQ.1) THEN

            X = XLOWS

            Y = YLOWS + RRAA1*(YHIGHS-YLOWS)
            Z = ZLOWS + RRAA2*(ZHIGHS-ZLOWS)
          END IF

      END IF
      END IF                                                            HIST  00212
C                                                                       HIST  00213
      XB(1) = X                                                         HIST  00220
      XB(2) = Y                                                         HIST  00221
      XB(3) = Z                                                         HIST  00222
      WT(1) = STH(1)*CPH(1)                                             HIST  00223
      WT(2) = STH(1)*SPH(1)                                             HIST  00224
      WT(3) = CTH(1)                                                    HIST  00225
C     ------------                                                      HIST  00226
      CALL ZONEA                                                        HIST  00227
C     ------------                                                      HIST  00228
      LB   = IR                                                         HIST  00229
      LBCZ = IRPRIM                                                     HIST  00230
      IPR = 1                                                           HIST  00232
C                                                                       HIST  00233
C     -------------------------                                         HIST  00234
C ... CALL TRACKING ROUTINES                                            HIST  00235
C     -------------------------                                         HIST  00236
   70 IF ( FLESRC .OR. (IPR .NE. 1) ) THEN                              HIST  00237
C       -------------------------------------                           HIST  00238
C ...   PARTICLE TO BE TRACKED IS AN ELECTRON                           HIST  00239
C       -------------------------------------                           HIST  00240
        IF (MT .NE. MAT(LB)) THEN                                       HIST  00241
          MT = MAT(LB)                                                  HIST  00242
        END IF                                                          HIST  00248
C         ------------                                                  HIST  00249
          CALL EHIST                                                    HIST  00250
C         ------------                                                  HIST  00251
      ELSE                                                              HIST  00252
C       ---------------------------------                               HIST  00253
C  ...  PARTICLE TO BE TRACKED IS A PHOTON                              HIST  00254
C       ---------------------------------                               HIST  00255
        LPCZ = LBCZ                                                     HIST  00262
C       ------------------------------------------------------          HIST  00265
        CALL PHIST(X,Y,Z,LB,CTH(1),STH(1),CPH(1),SPH(1),T,W,1)          HIST  00266
C       ------------------------------------------------------          HIST  00267
      END IF                                                            HIST  00269
C                                                                       HIST  00270
```

```
C  ______________________________________________________________     HIST     00271
C ... REMOVE SECONDARY ELECTRONS FROM STORAGE FOR TRANSPORT            HIST     00272
C ---------------------------------------------------------------     HIST     00273
        IF (LAST .NE. 0) THEN                                         HIST     00274
          LB      = LBS(LAST)                                         HIST     00275
          Z       = ZS(LAST)                                          HIST     00276
          T       = TS(LAST)                                          HIST     00277
          NT      = NTS(LAST)                                         HIST     00278
          CTH(1)  = CTHS(LAST)                                        HIST     00279
          W       = WS(LAST)                                          HIST     00280
          IPR     = IPRS(LAST)                                        HIST     00281
C                                                                     HIST     00283
          X       = XS(LAST)                                          HIST     00284
          Y       = YS(LAST)                                          HIST     00285
          STH(1)  = STHS(LAST)                                        HIST     00286
          CPH(1)  = CPHS(LAST)                                        HIST     00287
          SPH(1)  = SPHS(LAST)                                        HIST     00288
C                                                                     HIST     00289
          LBCZ  = LBCS(LAST)                                          HIST     00291
          KLOOP = KLOOP+1                                             HIST     00292
          LAST = LAST-1                                               HIST     00294
          GO TO 70                                                    HIST     00295
        END IF                                                        HIST     00296
C                                                                     HIST     00297
        IF (.NOT. FLPHD) GO TO 1299                                   HIST     00298
C                                                                     HIST     00299
C  ____________________________________                               HIST     00300
C ... SCORE PULSE-HEIGHT DISTRIBUTION                                 HIST     00301
C ------------------------------------                                HIST     00302
        EABST = CZERO                                                 HIST     00303
        DO 100 LS=LPHDB,LPHDE                                         HIST     00304
          EABST      =  EABST+PHDD(LS)                                HIST     00305
  100     PHDD(LS)  = CZERO                                           HIST     00306
        DO 110 JS=1,JSMAX                                             HIST     00307
          IF(SMARK(JS) .LE. EABST) GO TO 120                          HIST     00308
  110   CONTINUE                                                      HIST     00309
        NPHD = NPHD+1                                                 HIST     00310
        GO TO 1299                                                    HIST     00311
  120   ABE(JS)  = ABE(JS)+CWCF                                       HIST     00312
 1299   IF(NINDV.EQ.0)GO TO 130
        DO 1298 NIND=1,NINDV                                          New
         EDTL(NIND)=EDPR(NIND)+EDNK(NIND)+EDSC(NIND)                  Code
1298  CONTINUE
          WRITE(44)(EDPR(NIND),EDNK(NIND),EDSC(NIND),EDTL(NIND),NIND
     $ =1,NINDV)
           130 CONTINUE                                                   HIST     00313
C        __________________                                           HIST     00314
       CALL RANSAV(IRC)                                               HIST     00315
C      ------------------                                             HIST     00316
       RETURN                                                         HIST     00317
       END                                                            HIST     00318
```

```
      SUBROUTINE SRCINF(IMXOLD)                                         SRCINF    00002
C ***********************************************************************SRCINF    00003
C                                                                       SRCINF    00004
C     PROGRAM SRCINF IS CALLED BY                                       SRCINF    00005
C                                INPUT                                  SRCINF    00006
C     PROGRAM INPUT CALLS                                               SRCINF    00007
C        INTRINSIC FUNCTIONS                                            SRCINF    00008
C                             MAX,  SIN,  COS,   SQRT                   SRCINF    00009
C        EXTERNAL FUNCTIONS                                             SRCINF    00010
C                             ABORTX                                    SRCINF    00011
C                                                                       SRCINF    00012
C     ORIGINATION DATE      28 NOV 89                                   SRCINF    00013
C     LAST MODIFIED         11 MARCH 91                                 SRCINF    00014
C                                                                       SRCINF    00015
C     FUNCTION                                                          SRCINF    00016
C        This Subroutine processes the SOURCE Information               SRCINF    00017
C                                                                       SRCINF    00018
C     INPUT PARAMETERS                                                  SRCINF    00019
C        IMXOLD  -                                                      SRCINF    00020
C                                                                       SRCINF    00021
C     OUTPUT PARAMETERS                                                 SRCINF    00022
C        NONE                                                           SRCINF    00023
C                                                                       SRCINF    00024
C ***********************************************************************SRCINF    00025
C *** COMMON BLOCKS CNSTNT, PARAMS, OUT, CALC, STTS                     SRCINF    00026
C$    LIST(S=0)                                                         SRCINF    00027
CDIR$ NOLIST                                                            SRCINF    00028
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               CNSTNT    00081
```

No changes in CNSTNT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
                                                                        CNSTNT    00082
C                                                                       PARAMS    00002
C --------------------------------------------------------------------- PARAMS    00003
C                                                                       PARAMS    00004
```

PARAMS common block identical to that shown in subroutine INPUT

```
C ... I/O UNIT DECLARATIONS AND ARRAY BUFFERS                           PARAMS    00005
      PARAMETER (IIN = 5, IOUT = 6, ITP10 = 10, ITP11 = 11, ITP12 = 12, PARAMS    00006
     $           ITP14 = 14, MAXKEY = 36)
C                                                                       PARAMS    00008
        COMMON /EXTSORC/ IRECTS,IDISKS,XLOWS,XHIGHS,YLOWS,YHIGHS,ZLOWS,
     $  ZHIGHS,XCENT,YCENT,ZCENT,XCIR,YCIR,ZCIR,KPERPYZ,KPREPXZ,KPERPXY
      LOGICAL RRKILL, FLMTEL                                            OUT       00002
      COMMON /OUT/                                                      OUT       00003
     1 FLMTEL(INGP)                                                     OUT       00004
C                                                                       OUT       00005
```

New Code

No changes in OUT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C                                                                       CALC      00002
      COMMON /CALC/                                                     CALC      00003
     1 ACON(INMT),          ASTEP(INMAX,INMT),       AT(NSURV,INMT),    CALC      00004
```

No changes in CALC common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C                                                                       CALC      00139
      LOGICAL DMPFLG, FLMC                                              STTS      00002
      DOUBLE PRECISION IRSAV                                            STTS      00010
      COMMON  /STTS/  IB,  NB,  NSORS, IBT, BOLD, BATCH, KPUTMX, DMPFLG STTS      00017
     $ ,  IHIST,   IRSAV,  KPUT, FLMC                                   STTS      00018
C                                                                       STTS      00019
C$    LIST(S=1)                                                         SRCINF    00034
CDIR$ LIST                                                              SRCINF    00035
C                                                                       SRCINF    00036
      WRITE(IOUT,'(''1*************************''/                      SRCINF    00037
     $             '' *  SOURCE INFORMATION  *''/                       SRCINF    00038
     $             '' *************************'')')                    SRCINF    00039
C                                                                       SRCINF    00040
```

```
      IF (FLESRC) THEN                                                  SRCINF   00041
        WRITE(IOUT,'(''0SOURCE ELECTRONS'')')                           SRCINF   00042
      ELSE                                                              SRCINF   00043
        WRITE(IOUT,'(''0SOURCE PHOTONS'')')                             SRCINF   00044
      END IF                                                            SRCINF   00045
C                                                                       SRCINF   00046
      WRITE(IOUT,'(''0THE MAXIMUM SOURCE ENERGY IS'',T38,F12.5,         SRCINF   00047
     $             '' MEV'')') TIN                                      SRCINF   00048
      WRITE(IOUT,'(''0THE GLOBAL ELECTRON CUTOFF ENERGY IS'',T38,F12.5, SRCINF   00049
     $             '' MEV'')') TCUT                                     SRCINF   00050
      WRITE(IOUT,'(''0THE PHOTON CUTOFF ENERGY IS'',T38,F12.5,          SRCINF   00051
     $             '' MEV'')') TPCUT                                    SRCINF   00052
      IF (TSAVE .GT. TCUT) WRITE(IOUT,'(''0THE GLOBAL ELECTRON TRAP'',  SRCINF   00053
     $             ''PING ENERGY IS'',T38,F12.5,'' MEV'')') TSAVE       SRCINF   00054
C                                                                       SRCINF   00055
      IF (FLSPEC) THEN                                                  SRCINF   00056
        WRITE(IOUT,'(''0SOURCE SPECTRUM'')')                            SRCINF   00057
        WRITE(IOUT,'(12I6)') JSPEC                                      SRCINF   00058
        WRITE(IOUT,'(''0NORMALIZED CUMULATIVE SPECTRUM'')')             SRCINF   00059
        WRITE(IOUT,'(6F12.5)') (SPECIN(J),J=1,JSPEC)                    SRCINF   00060
        IF ((SPECIN(1) .NE. CONE) .OR. (SPECIN(JSPEC) .NE. CZERO)) THEN SRCINF   00061
          WRITE(IOUT,*) ' INPUT CUMULATIVE SOURCE SPECTRUM MUST BE',    SRCINF   00062
     $                  ' MONOTONICALLY DECREASING FROM 1.0 TO 0.0'     SRCINF   00063
C        _______________________                                        SRCINF   00064
          CALL ABORTX('SRCINF')                                         SRCINF   00065
C        -----------------------                                        SRCINF   00066
        END IF                                                          SRCINF   00067
        WRITE(IOUT,'(''0SPECTRAL ENERGIES (MEV)'')')                    SRCINF   00068
        WRITE(IOUT,'(6F12.5)') (ESP(J),J=1,JSPEC)                       SRCINF   00069
C                                                                       SRCINF   00070
      END IF                                                            SRCINF   00071
C
      IF(IRECTS.EQ.0 .AND. IDISKS.EQ.0) THEN
```

New Code

```
C                                                                       SRCINF   00072
      WRITE(IOUT,'(''0COORDINATES OF THE POINT SOURCE OR OF THE'',      SRCINF   00083
     $             '' CENTER OF THE BEAM (DISK) SOURCE ARE''/           SRCINF   00084
     $             ''     X = '',E12.5,'' CM'',10X,''Y = '',E12.5,      SRCINF   00085
     $             '' CM'',10X,''Z = '',E12.5,'' CM'')')                SRCINF   00086
     $              XSR, YSR, ZSR                                       SRCINF   00087
      WRITE(IOUT,'(''0THE RADIUS OF THE BEAM (DISK) SOURCE IS = '',     SRCINF   00088
     $              1PE12.4,'' CM'')') SORCIN                           SRCINF   00089
C                                                                       SRCINF   00095
      END IF
```

New Code

```
C
      WRITE(IOUT,'(''0REFERENCE DIRECTION FOR ANGULAR DISTRIBUTION'',   SRCINF   00090
     $             '' IS DEFINED BY''/''     THETA = '',G11.4,          SRCINF   00091
     $             '' DEGREES'',10X,''PHI = '',G11.4,'' DEGREES'')')    SRCINF   00092
     $              CTSR, CPSR                                          SRCINF   00093

      TEMPA = CTSR/C180PI                                               SRCINF   00096
      CTSR  = COS(TEMPA)                                                SRCINF   00097
      STSR  = SIN(TEMPA)                                                SRCINF   00099
      TEMPA = CPSR/C180PI                                               SRCINF   00100
      CPSR  = COS(TEMPA)                                                SRCINF   00101
      SPSR  = SIN(TEMPA)                                                SRCINF   00102
C
      IF(IRECTS.EQ.1)THEN
          WRITE(IOUT,55)XLOWS,XHIGHS,YLOWS,YHIGHS,ZLOWS,ZHIGHS
55    FORMAT(//1X,'RECTANGULAR PLATE SOURCE,'/5X,'BOUNDING COORDINATES A
     $RE - '/10X,'XLOW = ',E12.5,'    XHIGH = ',E12.5/10X,'YLOW = ',E12.5
     $,'    YHIGH = ',E12.5/10X,'ZLOW = ',E12.5,'    ZHIGH = ',E12.5)
          IF(ABS(XHIGHS-XLOWS).LE.CT1EM7)KPERPYZ=1
          IF(ABS(YHIGHS-YLOWS).LE.CT1EM7)KPERPXZ=1
          IF(ABS(ZHIGHS-ZLOWS).LE.CT1EM7)KPERPXY=1
               KPRPSUM=KPERPYZ+KPERPXZ+KPERPXY
               IF(KPRPSUM.EQ.0 .OR. KPRPSUM.EQ.3)THEN
               WRITE(IOUT,54)
54    FORMAT(///1X,'PROBLEM IN DEFINITION OF SOURCE PLANE ORIENTATION')
            CALL ABORTX('SRCINF')
               END IF
          END IF
```

New Code

```
C
      IF(IDISKS.EQ.1)THEN
         WRITE(IOUT,56)XCENT,YCENT,ZCENT,XCIR,YCIR,ZCIR
56    FORMAT(//1X,'CIRCULAR DISK SOURCE'/5X,'COORDINATES OF CENTER ARE'/
     $10X,'XCENTER = ',E12.5,'    YCENTER = ',E12.5,'    ZCENTER = ',E12.5
     $//5X,'COORDINATES OF POINT ON CIRCUMFERENCE ARE'/10X,'XCIR = ',
     $E12.5,'    YCIR = ',E12.5,'    ZCIR = ',E12.5)
         IF(ABS(XCENT-XCIR).LE.CT1EM7)KPERPYZ=1
         IF(ABS(YCENT-YCIR).LE.CT1EM7)KPERPXZ=1
         IF(ABS(ZCENT-ZCIR).LE.CT1EM7)KPERPXY=1
              KPRPSUM=KPERPYZ+KPERPXZ+KPERPXY
              IF(KPRPSUM.EQ.0 .OR. KPRPSUM.EQ.3)THEN
              WRITE(IOUT,54)
           CALL ABORTX('SRCINF')
              END IF

C

                 IF(SORCIN.EQ.CZERO)THEN

                 WRITE(IOUT,57)

                 CALL ABORTX('SRCINF')

                 END IF

C

            RSSQQ=SQRT((XCENT-XCIR)**2+(YCENT-YCIR)**2+(ZCENT-ZCIR)**2)
                 IF(ABS(RSSQQ-SORCIN).GT.CT1EM7)THEN

                 WRITE(IOUT,58)

                 CALL ABORTX('SRCINF')

                 ELSE

                 SORCIN=RSSQQ
                 WRITE(IOUT,59)SORCIN
59    FORMAT(/1X,'RADIUS OF THE DISK SOURCE IS ',E12.5)

                 END IF

57    FORMAT(///1X,'SOURCE DISK RADIUS NOT SPECIFIED')
58    FORMAT(///1X,'SOURCE DISK RADIUS INCONSISTENT WITH SPECIFICATION O
     $F POINTS ON CIRCUMFERENCE AND AT CENTER OF SOURCE DISK')


      END IF
C
C
C     This code block checks to see if the sine of the input polar
C     angle direction is less than zero. If it is, it allows this
C     condition within an acceptable tolerance and changes the sine
C     of the angle to zero; else it aborts.
C
      IF (STSR .LT. CZERO) THEN
        IF (STSR .GT. -C1EM6) THEN
          WRITE (IOUT,'(//,'' >>>> SRCINF: WARNING! SINE OF SOURCE'',
     $           '' INPUT POLAR ANGLE DIRECTION IS CHANGED TO ZERO.'')')
          STSR = CZERO
          CTSR = SIGN(CONE,CTSR)
        ELSE
          WRITE (IOUT,'(//,'' >>>> THE INPUT POLAR ANGLE WITH THE'',
     $           '' DIRECTION KEYWORD MUST BE BETWEEN ZERO AND 180'',
     $           '' DEGREES.'')')
C        ----------------------
          CALL ABORTX('SRCINF')
C        ----------------------
        END IF
      END IF
```

New Code

```
C
C
```

New Code

```
      IF(IRECTS.EQ.0 .AND. IDISKS.EQ.0) THEN
      WSRX = STSR*CPSR
      WSRY = STSR*SPSR
      WSRZ = CTSR
C     --------------------------------
C ... USUALLY W1(V) = R(V) X OMEGA(V)
C     --------------------------------
      W1X  = YSR*WSRZ - ZSR*WSRY
      W1Y  = ZSR*WSRX - XSR*WSRZ
      W1Z  = XSR*WSRY - YSR*WSRX
      XNRM = W1X*W1X + W1Y*W1Y + W1Z*W1Z
C     --------------------------
C ... UNLESS R(V) X OMEGA(V) = 0
C     --------------------------
      IF (XNRM .EQ. CZERO) THEN
C       -------------------------------------
C ...     IF I(V) * OMEGA(V) = 0, W1(V) = I(V)
C       -------------------------------------
        IF (WSRX .EQ. CZERO) THEN
          W1X = CONE
          W1Y = CZERO
          W1Z = CZERO
C         -------------------------------------
C ...       IF J(V) * OMEGA(V) = 0, W1(V) = J(V)
C         -------------------------------------
        ELSE IF (WSRY .EQ. CZERO) THEN
          W1X = CZERO
          W1Y = CONE
          W1Z = CZERO
C         -------------------------------------
C ...       IF K(V) * OMEGA(V) = 0, W1(V) = K(V)
C         -------------------------------------
        ELSE IF (WSRZ .EQ. CZERO) THEN
          W1X = CZERO
          W1Y = CZERO
          W1Z = CONE
C         ----------------------------------------
C ...       OTHERWISE, W1(V) = +OR- K(V) X OMEGA(V)
C         ----------------------------------------
        ELSE
          W1Z = CZERO
          W1X = CONE/SQRT(CONE + (WSRX/WSRY)**2)
          W1Y = -W1X*WSRX/WSRY
        END IF
      ELSE
        XNRM = SQRT(XNRM)
        W1X  = W1X/XNRM
        W1Y  = W1Y/XNRM
        W1Z  = W1Z/XNRM
      END IF
C
      W2X = WSRY*W1Z - WSRZ*W1Y
      W2Y = WSRZ*W1X - WSRX*W1Z
      W2Z = WSRX*W1Y - WSRY*W1X
      XSR = XSR + CT1EM7*WSRX
      YSR = YSR + CT1EM7*WSRY
      ZSR = ZSR + CT1EM7*WSRZ
      IF (SORCIN .EQ. CZERO) THEN
        XSR = XSR+CT1EM7*W1X
        YSR = YSR+CT1EM7*W1Y
        ZSR = ZSR+CT1EM7*W1Z
      END IF
C
      END IF
```

New Code

```
C
      IF (ICTH .EQ. 1) THEN
        WRITE(IOUT,'(''0MONODIRECTIONAL SOURCE IN REFERENCE'',
     $              '' DIRECTION'')')
      ELSE IF (ICTH .EQ. 2) THEN
        WRITE(IOUT,'(''0ISOTROPIC SOURCE TRUNCATED AT '',G11.4,
     $        '' DEGREES WITH RESPECT TO REFERENCE DIRECTION'')') CTHIN
        CTHIN = COS(CTHIN/C180PI)
      ELSE
        WRITE(IOUT,'(''0COSINE-LAW SOURCE TRUNCATED AT '',G11.4,
     $        '' DEGREES WITH RESPECT TO REFERENCE DIRECTION'')') CTHIN
        CTHIN = COS(CTHIN/C180PI)**2
      END IF
C
      IF (NB .LE. 0)       NB = 10
      IF (IMAX .LT. NB) IMAX = NB
      IMAX = IMAX/NB
C
```

```
      IF ((IBT .NE. 0) .AND. (IMAX .NE. IMXOLD)) THEN
C
C ... BATCH SIZES INCONSISTENT ON RESTART - TERMINATE RUN
C
        WRITE(IOUT,'(''0*** FATAL ERROR ON ATTEMPTED RESTART ***''/
     $   '' NEW BATCH SIZE = '',I10,'' DOESNT EQUAL OLD BATCH SIZE = '',
     $   I10/'' BATCH SIZES MUST MATCH TO CORRECTLY ACCUMULATE'',
     $   '' STATISTICS'' )') IMAX, IMXOLD
C
        CALL ABORTX('SRCINF')
C
      END IF
C
      NB = NB + IBT
      WRITE(IOUT,'(''0THE STANDARD ERROR ESTIMATES ARE BASED ON '',I5,
     $               '' BATCHES OF '',I7,'' HISTORIES EACH'')') NB,IMAX
C
      RETURN
C
      END
C
```

```
      SUBROUTINE KEYMAP(INDX,FLDUP)                                     KEYMAP   00003
C ***********************************************************************KEYMAP   00004
C                                                                       KEYMAP   00005
C     SUBROUTINE KEYMAP IS CALLED BY                                    KEYMAP   00006
C                                INPUT                                  KEYMAP   00007
C     SUBROUTINE KEYMAP CALLS                                           KEYMAP   00008
C         INTRINSIC FUNCTIONS                                           KEYMAP   00009
C                                                                       KEYMAP   00010
C         EXTERNAL FUNCTIONS                                            KEYMAP   00011
C                                                                       KEYMAP   00012
C                                                                       KEYMAP   00013
C     ORIGINATION DATE      15 AUG 90                                   KEYMAP   00014
C     LAST MODIFIED         11 MARCH 91                                 KEYMAP   00015
C                                                                       KEYMAP   00016
C     FUNCTION                                                          KEYMAP   00017
C        This subroutine contains the INPUT Primary Keyword mapping.    KEYMAP   00018
C        It takes the "indx" of the keyword list array as input and     KEYMAP   00019
C        returns the status of the duplicate keyword flag, "fldup".     KEYMAP   00020
C                                                                       KEYMAP   00021
C     INPUT PARAMETERS                                                  KEYMAP   00022
C        INDX    - Index of the keyword list array                      KEYMAP   00023
C                                                                       KEYMAP   00024
C     OUTPUT PARAMETERS                                                 KEYMAP   00025
C        FLDUP    - Status of the duplicate keyword flag                KEYMAP   00026
C                                                                       KEYMAP   00027
C ***********************************************************************KEYMAP   00028
C *** COMMON BLOCKS CNSTNT, PARAMS                                      KEYMAP   00029
C$    LIST(S=0)                                                         KEYMAP   00030
CDIR$ NOLIST                                                            KEYMAP   00031
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               CNSTNT   00081
```

No changes in CNSTNT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
                                                                        CNSTNT   00082
C                                                                       PARAMS   00002
C --------------------------------------------------------------------- PARAMS   00003
C                                                                       PARAMS   00004
```

PARAMS common block identical to that shown in subroutine INPUT

```
C$    LIST(S=1)                                                         KEYMAP   00034
CDIR$ LIST                                                              KEYMAP   00035
C                                                                       KEYMAP   00036
      CHARACTER*17 OKEYLS(MAXKEY)                                       KEYMAP   00037
      LOGICAL FLDUP                                                     KEYMAP   00038
C                                                                       KEYMAP   00039
      DATA OKEYLS /'BATCHES', 'CUTOFFS',                                KEYMAP   00040
     $             'DIRECTION', 'DUMP', 'ECHO',                         KEYMAP   00041
     $             'ELECTRONS', 'ELECTRON-ESCAPE', 'ELECTRON-FLUX',     KEYMAP   00042
     $             'ENERGY', 'GEOMETRY', 'HISTORIES', 'NEW-DATA-SET',   KEYMAP   00043
     $             'NEXT-EVENT-ESCAPE', 'NO-KNOCKONS', 'NO-STRAGGLING', KEYMAP   00044
     $             'PHOTONS', 'PHOTON-ESCAPE', 'PHOTON-FLUX', 'PLOTS',  KEYMAP   00045
     $             'POSITION', 'PRINT-ALL', 'PULSE-HEIGHT',             KEYMAP   00046
     $             'RANDOM-NUMBER', 'RESTART', 'SCALE-BREMS',           KEYMAP   00047
     $             'SCALE-IMPACT', 'SIMPLE-BREMS', 'SPECTRUM', 'TITLE', KEYMAP   00048
     $             'TRAP-ELECTRONS', 'NO-COHERENT', 'NO-INCOH-BINDING', KEYMAP   00049
C    $             'DETAIL-IONIZE'/                                     KEYMAP   00050
     $             'DETAIL-IONIZE','RECTANGLE-SOURCE','CIRCLE-SOURCE',
     $             'INDIVIDUAL-HISTS'/
     C                                                                  KEYMAP   00051
C     Print that the keyword pointed to by INDX is a duplicate entry    KEYMAP   00052
C                                                                       KEYMAP   00053
      WRITE(IOUT,'('' >>>> KEYMAP: DUPLICATE INPUT KEYWORD: '', (A))')  KEYMAP   00054
     $      OKEYLS(INDX)                                                KEYMAP   00055
      IF (.NOT. FLDUP) FLDUP = .TRUE.                                   KEYMAP   00056
C                                                                       KEYMAP   00057
      RETURN                                                            KEYMAP   00058
      END                                                               KEYMAP   00059
```

New Code

APPENDIX 3

count.F Program Listing

```
      parameter (maxhis = 10000)
      parameter (mxclls = 2)
      dimension edep(200),nelec(200),nprot(200),nneut(200),nphot(200)
     1,edelec(200),edprot(200),edphot(200),edneut(200),iesc(10),

     2numcoin(3),koinc(3,mxclls),kcell(mxclls)
      dimension nhcoin(maxhis,3),npart(3)
      data npart/9,1,2/
      data kcell/67,88/
      data eps/1.e-8/
         data ntime,nw8win,nclimp,nwcut,nh,iesc,numcoin/18*0/
      data eesc,elostot/0.0,0.0/
      data edeptot,eprtot,ephtot,edntot,edeltot/5*0./
      data edep,edprot,edphot,edelec,edneut/1000*0./
      data nelec,nprot,nneut,nphot/800*0/
      open(1,file='trkbin',status='unknown',form='UNFORMATTED')
      open(7,file='countups',status='unknown')
         print 77
         read(5,*)nmax
77    format(1x,'Enter number of histories')
 111  format(1x,'problem in track file')
c     start a history
      do 18 i=1,3
      do 18 k=1,mxclls
 18   koinc(i,k)=0
 20   nh=nh+1
      lflag=0
         elost=0.0
      if(nh.gt.nmax)go to 2000
      read(1,end=2000)nhstry,nstart
      read(1)nevent,nsrc,ipt,ncell,mat,xs,ys,zs,us,vs,ws,es,wt,time
      do 25 nc=1,mxclls
      do 25 i=1,3
       np=npart(i)
 25   if(ipt.eq.np .and. ncell.eq.kcell(nc)) koinc(i,nc)=koinc(i,nc)+1
      ec=es
      iptold=ipt
      ncold=ncell
      oldtim=time
c     read event records
 1000 read(1)nevent,nsurf,i1,ipt,ncell,mat,x,y,z,u,v,w,e,wt,time
      do 26 nc=1,mxclls
      do 26 i=1,3
       np=npart(i)
 26   if(ipt.eq.np .and. ncell.eq.kcell(nc)) koinc(i,nc)=koinc(i,nc)+1
      if(i1.eq.0.and.lflag.gt.0)elost=elost-e*wt
         nter=0
         de=ec-e
      dt=time-oldtim
      if(nevent.eq.9000)then
         nter=nsurf
```

```
        nbrnch=i1
      if  (nter.eq.1)then
      iesc(ipt)=iesc(ipt)+1
      eesc=eesc+e*wt/nmax

        go to 20
     end if
        if(nter.gt.2)then
        if(nter.eq.3)ntime=ntime+1
        if(nter.eq.4)nw8win=nw8win+1
        if(nter.eq.5)nclimp=nclimp+1
        if(nter.eq.6)nwcut=nwcut+1
        end if
      end if
        if(nevent.ge.2000. and. nevent.lt.3000)then
         nter=nsurf
        if(nter.gt.12.and.nter.lt.15)then
         lflag=lflag+1
         elost=elost+e
      end if
      if(nter.eq.1)then
      iesc(ipt)=iesc(ipt)+1
      eesc=eesc+e*wt/nmax
      go to 19
      end if
      end if
      if(nter.eq.2)de=ec-e
       if(ipt.eq.iptold .and. dt.ge.eps.and.de.ge.eps)then
c     energy deposition by charged particle CSDA or neutral particle
c     inelastic collision
            edep(ncold)=edep(ncold)+de*wt
c     electron
            if(ipt.eq.3)then
                 nelec(ncold)=nelec(ncold)+1
                 edelec(ncold)=edelec(ncold)+de*wt
            end if
c     proton
            if(ipt.eq.9)then
                nprot(ncold)=nprot(ncold)+1
                edprot(ncold)=edprot(ncold)+de*wt
            end if
c     neutron
            if(ipt.eq.1)then
                nneut(ncold)=nneut(ncold)+1
                edneut(ncold)=edneut(ncold)+de*wt
            end if
c     photon
            if(ipt.eq.2)then
               nphot(ncold)=nphot(ncold)+1
               edphot(ncold)=edphot(ncold)+de*wt
            end if
      end if
   19 iptold=ipt
      ec=e
      oldtim=time
      ncold=ncell
        if(nevent.eq.9000)then
```

```
        elostot=elostot+elost
        do 220 k=1,3
        do 219 nc=1,mxclls
 219    if(koinc(k,nc).eq.0)go to 220
          numcoin(k)=numcoin(k)+1
          ll=numcoin(k)
          nhcoin(ll,k)=nh
 220    continue
        do 221 k=1,3
        do 221 nc=1,mxclls
 221    koinc(k,nc)=0
           go to 20
           end if
      go to 1000
 2000 continue
      elostot=elostot/nmax
      do 2100 m=1,200
      edep(m)=edep(m)/nmax
      edprot(m)=edprot(m)/nmax
      edphot(m)=edphot(m)/nmax
      edneut(m)=edneut(m)/nmax
      edelec(m)=edelec(m)/nmax
         edeptot=edeptot+edep(m)
         eprtot=eprtot+edprot(m)
         ephtot=ephtot+edphot(m)
         edntot=edntot+edneut(m)
         edeltot=edeltot+edelec(m)
 2100 write(7,112)m,edep(m),nprot(m),edprot(m),nelec(m),edelec(m),
     1nneut(m),edneut(m),nphot(m),edphot(m)
         edeptot=edeptot+elostot
      write(7,119)(iesc(i),i=1,10),eesc
         write(7,114)edeptot,eprtot,ephtot,edntot,edeltot,elostot,
     $numcoin(1),numcoin(2),numcoin(3)
      do 300 k=1,3
      maxk=numcoin(k)
      if(k.eq.1)write(7,302)
      if(k.eq.2)write(7,303)
      if(k.eq.3)write(7,304)
      write(7,301)(nhcoin(m,k),m=1,maxk)
 300   continue
 301  format(15i8)
 302  format(/1x,'history numbers for proton coincidence events')
 303  format(/1x,'history numbers for neutron coincidence events')
 304  format(/1x,'history numbers for photon coincidence events')
 114  format(1x,'total energy deposited = ',e12.5/10x,'from protons = ',
     xe12.5,/10x,'from photons = ',e12.5/10x,'from neutrons = ',e12.5/
     x10x,'from electrons = ',e12.5,/10x,'from inelastic collisions='
     $,e12.5/10x,'number of proton coincidence events =',i5
     $/10x,'number of neutron coincidence events =',i5
     $/10x,'number of photon coincidence events =',i5)
 119  format(1x,'no. of escaped particles =',10i6,/1x,
     $' escaped energy =',e12.5)
 112  format(1x,i5,e12.5,4(i15,e12.5))
      stop
      end
```

APPENDIX 4

source.F Program Listing for MCNPX Beam Source Allowing for User-supplied Location, Direction, Energy, Particle Specie

```
c_deck so source                                                        so
1
      subroutine source                                                 so
2
c   user supplied source subroutine
#include "cm.h"
c
      data issty/0/
      if(issty.eq.0)then
      wgt=1.0
      tme=0.0
      write(jtty,1)
      read(itty,*)xxx,yyy,zzz
      write(jtty,2)
      read(itty,*)uuu,vvv,www
      aa=sqrt(uuu**2+vvv**2+www**2)
      uuu=uuu/aa
      vvv=vvv/aa
      www=www/aa
      write(jtty,7)
      read(itty,*)ipt
      write(jtty,3)
      read(itty,*)erg
      write(jtty,4)
      read(itty,*)jsu
      write(jtty,5)
      read(itty,*)icl
      write(47)wgt,tme,xxx,yyy,zzz,uuu,vvv,www,erg,ipt,jsu,icl
      print 6,xxx,yyy,zzz,uuu,vvv,www,icl,jsu,ipt,erg,wgt,tme
      write(iuo,8)
      write(iuo,6)xxx,yyy,zzz,uuu,vvv,www,icl,jsu,ipt,erg,wgt,tme
      write(iuo,9)
      issty=1
      else
      rewind 47
      read(47)wgt,tme,xxx,yyy,zzz,uuu,vvv,www,erg,ipt,jsu,icl
      do 50 ispr=1,3
 50   spare(ispr)=0.0
      end if
 1    format(1x,'Enter the source point (x,y,z)')
 2    format(1x,'Enter the source beam direction cosines (u,v,w)')
 3    format(1x,'Enter the source energy (MeV)')
 4    format(1x,'If this is a surface source, enter surface number'/
     $1x, 'if not, enter 0')
 5    format(1x,'Enter the cell number containing the source point')
 7    format(1x,'Enter particle type (ipt)')
 6    format(1x,'User-supplied source'/1x,'xxx = ',e12.5,' yyy = ',e12.5
     $,' zzz = ',e12.5/1x,'uuu = ',e12.5,' vvv = ',e12.5,' www = ',e12.5
     $/1x,'icl = ',i5,' jsu = ',i5,' ipt = ',i5/1x,'erg = ',e12.5,
     $' wgt = ',e12.5,' time = ',e12.5)
 8    format(/////1x,'*********************************************',
     $            1x,'*********************************************',
     $            1x,'*********************************************'/)
 9    format(/    1x,'*********************************************',
     $            1x,'*********************************************',
     $            1x,'*********************************************'/////)
      return
      end                                                               so
13
```

APPENDIX 5
ITS-ACCEPT Input File for the CEASE-DD1 Dosimeter

```
 10 MEV DOME SOURCE FLAT DOSIMETER TEST -DD1
********************** GEOMETRY **********************
GEOMETRY
*1
 RCC   0.0   0.0   -0.2032    0.0   0.00000  0.2032    1.75514
*2
 RCC   0.0   0.0   -0.36068   0.0    .00000  0.15748   1.75514
*3
 RCC   0.0   0.0   -0.36158   0.0    .00000  0.00090   1.75514
*4
 RCC   0.0   0.0   -0.38158   0.0    .00000  0.02000   1.75514
*5
 RCC   0.0   0.0   -0.45158   0.0    .00000  0.07000   1.75514
*6
 RCC   0.0   0.0   -0.55158   0.0    .00000  0.10000   1.75514
*7
 RCC   0.0   0.0   -0.70400   0.0    .00000  0.15242   1.75514
*8
 RCC   0.0   0.0   -1.20400   0.0    .00000  0.50000   1.75514
*9
 RCC   0.0   0.0   -0.2032    0.0   0.00000  0.2032    1.76
*10
 RCC   0.0   0.0   -0.36068   0.0    .00000  0.15748   1.76
*11
 RCC   0.0   0.0   -0.36158   0.0    .00000  0.00090   1.76
*12
 RCC   0.0   0.0   -0.38158   0.0    .00000  0.02000   1.76
*13
 RCC   0.0   0.0   -0.45158   0.0    .00000  0.07000   1.76
*14
 RCC   0.0   0.0   -0.55158   0.0    .00000  0.10000   1.76
*15
 RCC   0.0   0.0   -0.70400   0.0    .00000  0.15242   1.76
*16
 RCC   0.0   0.0   -1.20400   0.0    .00000  0.50000   1.76
*17
 RPP   -0.85  0.85    -0.85  0.85  -0.36158  -.36068
*18
 RPP   -0.6477  0.6477  -0.6477  0.6477  -0.38158  -0.36158
*19
 RPP   -0.85  0.85    -0.85  0.85  -0.38158  -.36158
*20
 RPP   -0.45  0.45    -0.45  0.45  -0.45158  -0.38158
*21
 RPP   -0.6477  0.6477  -0.6477  0.6477   -0.45158  -0.38158
*22
 RPP   -0.85  0.85    -0.85  0.85  -0.45158  -0.38158
*23
 RPP   -0.6342  0.6342  -0.6342  0.6342  -0.55158  -0.45158
*24
 RPP   -0.6477  0.6477  -0.6477  0.6477  -0.55158  -0.45158
*25
 RPP   -0.85     0.85    -0.85     0.85    -0.55158  -0.45158
*26
 RPP   -0.6477  0.6477  -0.6477  0.6477  -0.70400  -0.55158
*27
 RPP   -0.85     0.85    -0.85     0.85    -0.70400  -0.55158
*28
 SPH   0.0   0.0   0.0  1.75514
*29
 SPH   0.0   0.0   0.0  1.76
*30
 RCC   0.0   0.0   0.0        0.0   0.00000  1.76   1.75514
*31
 RCC   0.0   0.0   0.0        0.0   0.00000  1.76   1.76
*32
 SPH   0.0   0.0  0.0  5.0
*33
 SPH   0.0   0.0   0.0        10.0
```

```
  END

*VOID
  Z01 +1
  Z02 +2
  Z03 +17
  Z04 +3 -17
  Z05 +11 -3 -17
  Z06 +18
  Z07 +19 -18
  Z08 +4 -19 -18
  Z09 +12 -4 -19 -18
  Z10 +20
  Z11 +21 -20
  Z12 +22 -21 -20
  Z13 +5 -22  -21 -20
  Z14 +13 -5  -22 -21 -20
  Z15 +23
  Z16 +24 -23
  Z17 +25 -24 -23
  Z18 +6 -25 -24 -23
  Z19 +14 -6 -25 -24 -23
  Z20 +26
  Z21 +27 -26
  Z22 +7 -27 -26
  Z23 +15 -7 -27 -26
  Z24  +8
  Z25  +16 -8
  Z26  +28 +30
  Z27  +29 +31 -28
  Z28  +30 -29
  Z29  +9 -1
  Z30  +10 -2
  Z31 +31 -1 -2 -3 -4 -5 -6 -7 -8 -9 -10 -11 -12 -13 -14
      -15 -16 -17 -18 -19 -20 -21 -22 -23 -24 -25 -26 -26
      -28 -29 -30
  Z32 +32 -31
END
*MATERIAL
 1                                                    1
 0                                                    0
 1                                                    0
 0                                                    1
 0                                                    1
 0                                                    0
 1                                                    0
 0                                                    0
 0                                                    0
 2                                                    0
 0                                                    0
 1                                                    0
 0                                                    0
 0                                                    0
 3                                                    0
 0                                                    0
```

Two column format used here to conserve space

```
*********************** SOURCE ********************
ELECTRONS
SPECTRUM 11
  1.0000   .9553   .9062   .8559   .8046   .7508   .6932   .6312
   .5627   .4573   0.0
 10.0000  9.2000  9.1000  9.0000  8.9000  8.6000  8.3000  8.0000
  7.5000  5.0000  0.0
***********************  OPTIONS  *******************
PULSE-HEIGHT 10  10
   NBINE 102
DOME-SOURCE  0.  0.  0.0 1.755
***********************  OPTIONS  *******************
HISTORIES 10000
```

This is the hemispherical dome source option as depicted in Figure 15. For the flat disc source option depicted in Figure 14, this line must be replaced with

```
CIRCLE-SOURCE  0.  0.  -0.01 1.755  0.  -0.01
  RADIUS 1.755
     DIRECTION 180.0
ISOTROPIC
```

APPENDIX 6

MCNPX Input File for the CEASE-DD2 Dosimeter

```
CEASE DD2 dosimeter MCNPX, electrons, isotropic source on hemispherical void
c      Cells
c     Silicon Dosimeter DD2
1     2 -2.33 23 -25 22 -24 6 -5
c     Voids surrounding DD2
2     0 15 -17 14 -16 6 -5 #1
3     0 15 -17 14 -16 7 -6 #5
4     0 15 -17 14 -16 6 5 -4
c     Aluminum Oxide substrate
5     3 -3.97  19 -21 18 -20 7 -6
c     Aluminum base
6     1 -2.7 15 -17 14 -16 8 -7
c     Aluminum sides
7     1  -2.7  11 -13 10 -14 8 -4
8     1  -2.7  11 -13 16 -12 8 -4
9     1  -2.7  11 -15 14 -16 8 -4
10    1  -2.7  17 -13 14 -16 8 -4
c     Al foil
11    1  -2.7  11 -13 10 -12 4 -3
c     void cylinder above foil
12    0   -26 3 -2
c     Al cover plate
13    1  -2.7  2 -1 -26
c     Void cylinder around box
14    0  8 -3 -26 #1 #2 #3 #4 #5 #6 #7 #8 #9 #10 #11
c     Void cylinder below box
15    0  9 -8 -26
c     Void hemispherical region above plate
16    0  1 -27
c     Hemispherical void region enclosing upper half(dome)
17    0  27 -28 1
c     Spherical void region enclosing everything
18    0  -29  #1 #2 #3 #4 #5 #6 #7 #8 #9 #10 #11 #12 #13 #14 #15 #16
              #17
c     exterior void escape region
19    0     #18

c      Surfaces
1       pz  0.0
2       pz  -0.635
3       pz  -0.79248
4       pz  -0.79338
5       pz  -0.81338
6       pz  -0.88338
7       pz  -0.98338
8       pz  -1.1358
9       pz  -1.6358
10      py  -0.85
11      px  -0.85
12      py   0.85
13      px   0.85
14      py  -0.6477
15      px  -0.6477
16      py   0.6477
17      px   0.6477
18      py  -0.6342
19      px  -0.6342
20      py   0.6342
```

```
21      px   0.6342
22      py  -0.45
23      px  -0.45
24      py   0.45
25      px   0.45
26      cz  1.79578
27      so  1.79578
28      so  1.8
29      so  3.0
c

c    Transport electrons and photons
mode    e  p
c    Source  is defined by subroutine "sourcedd2.F" (which must be renamed
c           to "source.F")
c   no. of histories
nps     10000
c
c      electron cutoff energy = 0.5 MeV
cut:e       1.e+10  0.5
c      photon cutoff energy = 10 keV
cut:p       1.e+10  0.01
c
c     materials
c
c     aluminum
M1      13027 -1.0
C       Silicon
M2      14000 -1.0
C       Aluminum Oxide
M3      13027  -0.529251  8016  -0.470749
C       Kel-F  (chlorotrifluoroethylene  C2ClF3)
M4      6000  -0.20625  17000  -0.30440  9019  -0.48935
c
c      maximum electron energy(MeV) needed for cross sections
phys:e  12.
c      maximum photon energy(MeV) needed for cross sections
phys:p  12.
c   tallies
c     energy deposition (MeV) tally
*F18:e    1  5 13
c    pulse height tally
F48:e     1  5 13
E48      0 1.e-5 .1 97I 9.9 9.99999 10.
c      cell importances for electrons
imp:e   1 17R  0
c      cell importances for photons
imp:p   1 17R 0
```

APPENDIX 7

ITS-ACCEPT Subroutine Modifications for Dome Source Option - Code Listings -

```
      SUBROUTINE INPUT                                                  INPUT    00007
C ***********************************************************************INPUT    00009
C                                                                       INPUT    00010
C     PROGRAM INPUT IS CALLED BY                                        INPUT    00011
C                             ITS                                       INPUT    00012
C     PROGRAM INPUT CALLS                                               INPUT    00013
C         INTRINSIC FUNCTIONS                                           INPUT    00014
C                               REAL         (TIGER & CYLTRAN)          INPUT    00015
C                               SQRT, ABS    (ACCEPT)                   INPUT    00016
C         EXTERNAL FUNCTIONS                                            INPUT    00017
C                               ALIST,  ELIST,  START,  PREP, KOP,      INPUT    00018
C                               REQALL, GEOMIN, SCRINF, OPOPTS          INPUT    00019
C                               KEYMAP, OPREAD                          INPUT    00020
C                               JOGEN   (ACCEPT)                        INPUT    00021
C                                                                       INPUT    00022
C     ORIGINATION DATE     12 DEC 67.                                   INPUT    00023
C     LAST MODIFIED        17 MAY 91                                    INPUT    00024
C                                                                       INPUT    00025
C     FUNCTION                                                          INPUT    00026
C        THIS PROGRAM IS USED TO READ AND PROCESS USER-SUPPLIED         INPUT    00027
C        CARD INPUT                                                     INPUT    00028
C                                                                       INPUT    00029
C ***********************************************************************INPUT    00030
C *** COMMON BLOCKS CNSTNT, PARAMS, OUT, CALC, XPED, STTS, SCALE, PLTITLINPUT    00031
C                   PAREM, GOMLOC (ACCEPT)                              INPUT    00032
C                   FLUOR         (PCODES)                              INPUT    00033
C                   PLOT          (PLOTS)                               INPUT    00034
C$    LIST(S=0)                                                         INPUT    00035
```

•
•
•

Code listing omitted here is identical to that given in Appendix 2

•
•
•

```
C$    LIST(S=1)                                                         INPUT    00051
CDIR$ LIST                                                              INPUT    00052
      COMMON  /SCALE/  BNUM, XNUM                                       SCALE    00002
        COMMON /EXTSORC/ IRECTS,IDISKS,XLOWS,XHIGHS,YLOWS,YHIGHS,ZLOWS,
     $  ZHIGHS,XCENT,YCENT,ZCENT,XCIR,YCIR,ZCIR,KPERPYZ,KPREPXZ,KPERPXY
     $  IDOME,RDOME
C                                                                       SCALE    00003
C                                                                       INPUT    00057
        COMMON /HITS/EDPR(10),EDNK(10),EDSC(10),EDTL(10),LHCL(10),NINDV
```

New Code

•
•

•

Code listing omitted here is identical to that given in Appendix 2

•
•
•

```
      NPRTCL = 1                                                    INPUT     00153
C
      IRECTS = 0
        IDISKS = 0
        KPERPYZ = 0
        KPERPXZ = 0
        KPERPXY = 0
        IDOME = 0
```

New Code

```
C
      NINDV=0
        DO 599 J=1,10
599   LHCL(J)=0
C                                                                   INPUT     00154
      TITLE   = ' '                                                 INPUT     00155
      NPRT    = 12                                                  INPUT     00156
      IECHO   = 0                                                   INPUT     00157
      NB      = 10                                                  INPUT     00158
      IMAX    = 1000                                                INPUT     00159
      IBT     = 0                                                   INPUT     00160
      MBSC    = 1                                                   INPUT     00161
      BOLD    = CZERO                                               INPUT     00162
      IMXOLD  = 0                                                   INPUT     00163
      INRAN   = CZERO                                               INPUT     00168
      BASE    = CTWO                                                INPUT     00170
      XNCYC   = CEIGHT                                              INPUT     00171
      TMFAC   = BASE**(-1.0/XNCYC)                                  INPUT     00172
      DMPFLG  = .FALSE.                                             INPUT     00173
C                                                                   INPUT     00174
C ... INITIALIZE LOGICALS FOR IDENTIFYING MATERIALS (NON-P CODES) OR INPUT     00175
C     ELEMENTS (P CODES) THAT ARE PRESENT IN A GIVEN PROBLEM - USED INPUT     00176
C     FOR IDENTIFYING RELEVANT LINE RADIATION.                      INPUT     00177
      NGP = NMT                                                     INPUT     00182
      DO 60 J=1,NGP                                                 INPUT     00184
   60   FLMTEL(J) = .FALSE.                                         INPUT     00185
C                                                                   INPUT     00186
      NPLOTS  = 0                                                   INPUT     00188
C    ------------------                                             INPUT     00193
C ... SOURCE VARIABLES                                              INPUT     00194
C    ------------------                                             INPUT     00195
      FLESRC = .TRUE.                                               INPUT     00196
      JSPEC  = 0                                                    INPUT     00197
      FLSPEC = .FALSE.                                              INPUT     00198
      TIN    = CONE                                                 INPUT     00199
      TPCUT  = C1EM2                                                INPUT     00200
      TCUT   = CZERO                                                INPUT     00201
      TSAVE  = CZERO                                                INPUT     00202
      ICTH   = 1                                                    INPUT     00203
      CTSR   = CZERO                                                INPUT     00204
      CTHIN  = C90                                                  INPUT     00205
      ZSR    = CZERO                                                INPUT     00210
      XSR    = CZERO                                                INPUT     00213
      YSR    = CZERO                                                INPUT     00214
      CPSR   = CZERO                                                INPUT     00215
      SORCIN = CZERO                                                INPUT     00216
C    --------------------------                                     INPUT     00218
C ... ELECTRON ESCAPE VARIABLES                                     INPUT     00219
C    --------------------------                                     INPUT     00220
```

```
      JMAX   = 10                                                        INPUT     00221
      FLESC = .FALSE.                                                    INPUT     00222
      ITMK   = 1                                                         INPUT     00223
      IAMK   = 1                                                         INPUT     00224
      KMAX   = 18                                                        INPUT     00225
      KMAZ   = 1                                                         INPUT     00226
      IAMKZ = 1                                                          INPUT     00228
C     ------------------------                                           INPUT     00230
C ... PHOTON ESCAPE VARIABLES                                            INPUT     00231
C     ------------------------                                           INPUT     00232
      JPMAX   = 10                                                       INPUT     00233
      FLESCP = .FALSE.                                                   INPUT     00234
      IPMK    = 1                                                        INPUT     00235
      IBMK    = 1                                                        INPUT     00236
      KPMAX   = 18                                                       INPUT     00237
      KPMAZ   = 1                                                        INPUT     00238
      IBMKZ   = 1                                                        INPUT     00240
C     ------------------------                                           INPUT     00242
C ... ELECTRON FLUX VARIABLES                                            INPUT     00243
C     ------------------------                                           INPUT     00244
      FLFLUX = .FALSE.                                                   INPUT     00245
      JFMAX  = 10                                                        INPUT     00246
      KFMAX  = 6                                                         INPUT     00247
      KFMAZ  = 1                                                         INPUT     00248
      IFAMKZ = 1                                                         INPUT     00250
      IFMK   = 1                                                         INPUT     00252
      IFAMK  = 1                                                         INPUT     00253
C     ----------------------                                             INPUT     00254
C ... PHOTON FLUX VARIABLES                                              INPUT     00255
C     ----------------------                                             INPUT     00256
      FLFLXP = .FALSE.                                                   INPUT     00257
      JFMAXP = 10                                                        INPUT     00258
      KFMAXP = 6                                                         INPUT     00259
      KFMAZP = 1                                                         INPUT     00260
      IFBMKZ = 1                                                         INPUT     00262
      IFMKP  = 1                                                         INPUT     00264
      IFBMK  = 1                                                         INPUT     00265
C     -----------------------------------                                INPUT     00266
C ... PULSE HEIGHT DISTRIBUTION VARIABLES                                INPUT     00267
C     -----------------------------------                                INPUT     00268
      FLPHD  = .FALSE.                                                   INPUT     00269
      JSMAX  = 12                                                        INPUT     00270
      IPHMK  = 1                                                         INPUT     00271
C                                                                        INPUT     00272
C     ***********************************************                   INPUT     00273
C     *              BEGIN READING INPUT                *                INPUT     00274
C     *  ZERO-LEVEL KEYWORDS IN ALPHABETICAL ORDER      *                INPUT     00275
C     ***********************************************                   INPUT     00276
C                                                                        INPUT     00280
C ... SET ERROR TRAP FLAG TO ZERO                                        INPUT     00281
      IERTRP = 0                                                         INPUT     00282
      NUMCRD = 0                                                         INPUT     00283
      FLNEWD = .FALSE.                                                   INPUT     00284
      FLDUP  = .FALSE.                                                   INPUT     00285
      DO 65 IKEY=1,MAXKEY                                                INPUT     00286
   65   FLKEY(IKEY) = .FALSE.                                            INPUT     00287
C                                                                        INPUT     00288
C ... READ THE NEXT CARD IN THE INPUT FILE                               INPUT     00289
C                                                                        INPUT     00290
C     ___________________________                                        INPUT     00291
   70 CALL OPREAD(1,IECHO,EOFLAG)                                        INPUT     00292
C     ---------------------------                                        INPUT     00293
C                                                                        INPUT     00294
C ... NOTE, COMMENT CARDS DENOTED BY * IN COLUMN 1, SKIPPED INTERNALLY   INPUT     00295
C                                                                        INPUT     00296
      IF (.NOT. EOFLAG) THEN                                             INPUT     00297
        NUMCRD = NUMCRD + 1                                              INPUT     00298
C                                                                        INPUT     00299
   80   IF (KOP('BATCHES') .GE. 1) THEN                                  INPUT     00300
C         ---------                                                      INPUT     00301
C ...       BATCHES                                                      INPUT     00302
```

```
C          ---------
C          Check if primary keyword has been used
C
           IKEY = 1
C
           IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                            ------------------------
           FLKEY(IKEY) = .TRUE.
C
           NB = PARM(1)
C
         ELSE IF (KOP('CUTOFFS') .GE. 0) THEN
C          ---------
C ...      CUTOFFS
C          ---------
           IKEY = 2
C
           IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                            ------------------------
           FLKEY(IKEY) = .TRUE.
C
           KARG = KOP('CUTOFFS')
           IF (KARG .GE. 1) THEN
             TCUT = PARM(1)
           END IF
           IF (KARG .GE. 2) THEN
             TPCUT = PARM(2)
           END IF
C
         ELSE IF (KOP('DETAIL-IONIZE') .GE. 0) THEN
C
C ...      DETAIL-IONIZATION
C          ------------------
           IKEY = 33
C
           IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                            ------------------------
           FLKEY(IKEY) = .TRUE.
C
           NPRTCL = 2
C
C
         ELSE IF (KOP('RECTANGLE-SOURCE').GE.0)THEN
C

C          RECTANGULAR PLANE SOURCE
C          ------------

           IKEY = 34
C

C
           IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                            ------------------------
           FLKEY(IKEY) = .TRUE.
C

           KARG = KOP('RECTANGLE-SOURCE')
         IF(KARG.LT.6)THEN
                WRITE(IOUT,68)

68     FORMAT(1X,'>>>>>')
                WRITE(IOUT,51)
                    WRITE(IOUT,68)

51     FORMAT(1X,' USER MUST ENTER 6 NUMBERS (XLOW,XHIGH,YLOW,YHIGH,ZLOW,
     $ZHIGH) TO DEFINE SOURCE LOWER AND UPPER COORDINATE LIMITS OF SOURC
     $E RECTANGLE')
            CALL ABORTX('INPUT')
         ELSE
            IRECTS = 1
```

```
         XLOWS = PARM(1)
         XHIGHS = PARM(2)
         YLOWS = PARM(3)
         YHIGHS = PARM(4)
         ZLOWS = PARM(5)
         ZHIGHS = PARM(6)
         END IF
C
       ELSE IF (KOP('CIRCLE-SOURCE').GE.0)THEN
C         ____________
C           CIRCLE PLANE SOURCE
C         ------------
           IKEY = 35
C
C                                ________________________
           IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                                ------------------------
           FLKEY(IKEY) = .TRUE.
C

           KARG = KOP('CIRCLE-SOURCE')
         IF(KARG.LT.6)THEN
                WRITE(IOUT,68)

                WRITE(IOUT,52)

                WRITE(IOUT,68)

52      FORMAT(1X,' USER MUST ENTER 6 NUMBERS - COORDINATES OF CIRCLE CENT
     $ER (XO,YO,ZO),AND COORDINATES A POINT ON CIRCUMFERENCE'/1X,'(XC,YC
     $,ZC) TO DEFINE POSITION AND ORIENTATION OF SOURCE CIRCLE')
C

             CALL ABORTX('INPUT')
         ELSE
             IDISKS = 1
         XCENT = PARM(1)
         YCENT = PARM(2)
         ZCENT = PARM(3)
         XCIR = PARM(4)
         YCIR = PARM(5)
         ZCIR = PARM(6)
         CALL OPREAD(1,IECHO,EOFLAG)
         IF(KOP('RADIUS').GE.1) THEN
         SORCIN = PARM(1)
         ELSE
         GO TO 80
         END IF

C

       END IF
C
C
       ELSE IF (KOP('INDIVIDUAL-HISTS').GE.0) THEN
C        __________________________________________
C         RECORD SINGLE HISTORY ENERGY DEPOSITIONS
C        ------------------------------------------
C
          IKEY = 36
C
          IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C
          FLKEY(IKEY) = .TRUE.
C
          KARG   = KOP('INDIVIDUAL-HISTS')
            IF(KARG.LT.1.OR. KARG.GT.10)THEN
```

```
                WRITE(IOUT,68)
                   WRITE(IOUT,688)
                   WRITE(IOUT,68)
688   FORMAT(1X,'USER MUST ENTER NO FEWER THAN 1 AND NO MORE THAN 10 CEL
     $L NUMBERS IN WHICH THE ENERGY DEPOSITION'/1X,'FOR INDIVIDUAL ELECT
     $RON HISTORIES ARE TO BE RECORDED.')
C
          CALL ABORTX('INPUT')
        ELSE
          DO 689 KRRG=1,KARG
689     LHCL(KRRG)=PARM(KRRG)
        NINDV=KARG
        WRITE(IOUT,587)
        WRITE(IOUT,588)(LHCL(KRRG),KRRG=1,NINDV)
588   FORMAT(1X,'ENERGY DEPOSITION FOR INDIVIDUAL HISTORIES WILL BE RECO
     $RDED ON FILE "EDSHOW.TXT" FOR CELL NOS.'/5X,10I5)
        WRITE(IOUT,587)
587   FORMAT(/1X,'*******************************************************
     $*************************************************************'
     $/1X,'*********************************************************
     $*******************************************************')
      END IF
C
      ELSE IF (KOP('DOME-SOURCE').GE.0)THEN
C        ____________
C         HEMISPHERICAL DOME SOURCE
C        ------------
          IKEY = 37
C
C                             _______________________
          IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)
C                          -----------------------
          FLKEY(IKEY) = .TRUE.
C
          KARG = KOP('DOME-SOURCE')
        IF(KARG.LT.4)THEN
              WRITE(IOUT,68)

              WRITE(IOUT,53)

              WRITE(IOUT,68)
                                                                New
53    FORMAT(1X,' USER MUST ENTER 4 NUMBERS - COORDINATES OF SPHERE CENT   Code
     $ER (XO,YO,ZO),AND SPHERE RADIUS'/1X,'(RDOME)')
C
            CALL ABORTX('INPUT')
        ELSE
            IDOME = 1
        XCENT = PARM(1)
        YCENT = PARM(2)
        ZCENT = PARM(3)
        RDOME = PARM(4)
C
      END IF
C

C

C
        ELSE IF (KOP('DIRECTION') .GE. 0) THEN                        INPUT    00344
C                                                                     INPUT    00345
C ...     DIRECTION                                                   INPUT    00346
```

```
C        -----------                                                    INPUT    00347
         IKEY = 3                                                       INPUT    00348
C                                                                       INPUT    00349
         IF (FLKEY(IKEY)) CALL KEYMAP(IKEY,FLDUP)                       INPUT    00350
C                         ------------------------                      INPUT    00351
         FLKEY(IKEY) = .TRUE.                                           INPUT    00352
C                                                                       INPUT    00353
```

•
•
•

Remaining portion of subroutine INPUT (omitted here for brevity) is identical to original ACCEPT [1] code

•
•
•

```
      END                                                               INPUT    01841
```

```
      SUBROUTINE KEYMAP(INDX,FLDUP)
C ********************************************************************
C
C     SUBROUTINE KEYMAP IS CALLED BY
C                              INPUT
C     SUBROUTINE KEYMAP CALLS
C         INTRINSIC FUNCTIONS
C
C         EXTERNAL FUNCTIONS
C
C
C     ORIGINATION DATE      15 AUG 90
C     LAST MODIFIED         11 MARCH 91
C
C     FUNCTION
C        This subroutine contains the INPUT Primary Keyword mapping.
C        It takes the "indx" of the keyword list array as input and
C        returns the status of the duplicate keyword flag, "fldup".
C
C     INPUT PARAMETERS
C        INDX     - Index of the keyword list array
C
C     OUTPUT PARAMETERS
C        FLDUP    - Status of the duplicate keyword flag
C
C ********************************************************************
C *** COMMON BLOCKS CNSTNT, PARAMS
C$    LIST(S=0)
CDIR$ NOLIST
```

No changes in CNSTNT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C
C ------------------------------------------------------------------
```

PARAMS common block identical to that shown in subroutine INPUT listed in Appendix 2.

```
C$    LIST(S=1)
CDIR$ LIST
C
      CHARACTER*17 OKEYLS(MAXKEY)
      LOGICAL FLDUP
C
      DATA OKEYLS /'BATCHES', 'CUTOFFS',
     $             'DIRECTION', 'DUMP', 'ECHO',
     $             'ELECTRONS', 'ELECTRON-ESCAPE', 'ELECTRON-FLUX',
     $             'ENERGY', 'GEOMETRY', 'HISTORIES', 'NEW-DATA-SET',
     $             'NEXT-EVENT-ESCAPE', 'NO-KNOCKONS', 'NO-STRAGGLING',
     $             'PHOTONS', 'PHOTON-ESCAPE', 'PHOTON-FLUX', 'PLOTS',
     $             'POSITION', 'PRINT-ALL', 'PULSE-HEIGHT',
     $             'RANDOM-NUMBER', 'RESTART', 'SCALE-BREMS',
     $             'SCALE-IMPACT', 'SIMPLE-BREMS', 'SPECTRUM', 'TITLE',
     $             'TRAP-ELECTRONS', 'NO-COHERENT', 'NO-INCOH-BINDING',
C    $             'DETAIL-IONIZE'/
     $             'DETAIL-IONIZE','RECTANGLE-SOURCE','CIRCLE-SOURCE',
     $             'INDIVIDUAL-HISTS','DOME-SOURCE'/
     C
C     Print that the keyword pointed to by INDX is a duplicate entry
C
      WRITE(IOUT,'('' >>>> KEYMAP: DUPLICATE INPUT KEYWORD: '', (A))')
     $      OKEYLS(INDX)
      IF (.NOT. FLDUP) FLDUP = .TRUE.
C
      RETURN
      END
```

New code

```
      SUBROUTINE SRCINF(IMXOLD)                                         SRCINF    00002
C **********************************************************************SRCINF    00003
C                                                                       SRCINF    00004
C     PROGRAM SRCINF IS CALLED BY                                       SRCINF    00005
C                                 INPUT                                 SRCINF    00006
C     PROGRAM INPUT CALLS                                               SRCINF    00007
C         INTRINSIC FUNCTIONS                                           SRCINF    00008
C                              MAX,  SIN,  COS,  SQRT                   SRCINF    00009
C         EXTERNAL FUNCTIONS                                            SRCINF    00010
C                              ABORTX                                   SRCINF    00011
C                                                                       SRCINF    00012
C     ORIGINATION DATE     28 NOV 89                                    SRCINF    00013
C     LAST MODIFIED        11 MARCH 91                                  SRCINF    00014
C                                                                       SRCINF    00015
C     FUNCTION                                                          SRCINF    00016
C        This Subroutine processes the SOURCE Information               SRCINF    00017
C                                                                       SRCINF    00018
C     INPUT PARAMETERS                                                  SRCINF    00019
C        IMXOLD  -                                                      SRCINF    00020
C                                                                       SRCINF    00021
C     OUTPUT PARAMETERS                                                 SRCINF    00022
C        NONE                                                           SRCINF    00023
C                                                                       SRCINF    00024
C **********************************************************************SRCINF    00025
C *** COMMON BLOCKS CNSTNT, PARAMS, OUT, CALC, STTS                     SRCINF    00026
C$    LIST(S=0)                                                         SRCINF    00027
CDIR$ NOLIST                                                            SRCINF    00028
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               CNSTNT    00081
      SAVE                                                              CNSTNT    00082
C
```

No changes in CNSTNT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C                                                                       PARAMS    00002
C ----------------------------------------------------------------------PARAMS    00003
```

PARAMS common block identical to that shown in subroutine INPUT listed in Appendix 2.

New code

```
        COMMON /EXTSORC/ IRECTS,IDISKS,XLOWS,XHIGHS,YLOWS,YHIGHS,ZLOWS,
     $  ZHIGHS,XCENT,YCENT,ZCENT,XCIR,YCIR,ZCIR,KPERPYZ,KPREPXZ,KPERPXY,
     $  IDOME,RDOME
      LOGICAL RRKILL, FLMTEL                                            OUT       00002
      COMMON /OUT/                                                      OUT       00003
     1 FLMTEL(INGP)                                                     OUT       00004
```

No changes in OUT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C                                                                       CALC      00002
      COMMON /CALC/                                                     CALC      00003
     1 ACON(INMT),          ASTEP(INMAX,INMT),      AT(NSURV,INMT),     CALC      00004
```

No changes in CALC common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C                                                                       CALC      00139
      LOGICAL DMPFLG, FLMC                                              STTS      00002
      DOUBLE PRECISION IRSAV                                            STTS      00010
      COMMON  /STTS/  IB,  NB,  NSORS, IBT, BOLD, BATCH, KPUTMX, DMPFLG STTS      00017
     $ ,  IHIST,   IRSAV,  KPUT, FLMC                                   STTS      00018
C                                                                       STTS      00019
C$    LIST(S=1)                                                         SRCINF    00034
CDIR$ LIST                                                              SRCINF    00035
C                                                                       SRCINF    00036
      WRITE(IOUT,'(''1**************************''/                     SRCINF    00037
     $               '' *  SOURCE INFORMATION  *''/                     SRCINF    00038
     $               '' **************************'')')                 SRCINF    00039
C                                                                       SRCINF    00040
      IF (FLESRC) THEN                                                  SRCINF    00041
        WRITE(IOUT,'(''0SOURCE ELECTRONS'')')                           SRCINF    00042
      ELSE                                                              SRCINF    00043
        WRITE(IOUT,'(''0SOURCE PHOTONS'')')                             SRCINF    00044
```

```
      END IF                                                            SRCINF   00045
C                                                                       SRCINF   00046
      WRITE(IOUT,'(''0THE MAXIMUM SOURCE ENERGY IS'',T38,F12.5,         SRCINF   00047
     $              '' MEV'')') TIN                                     SRCINF   00048
      WRITE(IOUT,'(''0THE GLOBAL ELECTRON CUTOFF ENERGY IS'',T38,F12.5, SRCINF   00049
     $              '' MEV'')') TCUT                                    SRCINF   00050
      WRITE(IOUT,'(''0THE PHOTON CUTOFF ENERGY IS'',T38,F12.5,          SRCINF   00051
     $              '' MEV'')') TPCUT                                   SRCINF   00052
      IF (TSAVE .GT. TCUT) WRITE(IOUT,'(''0THE GLOBAL ELECTRON TRAP'',  SRCINF   00053
     $              ''PING ENERGY IS'',T38,F12.5,'' MEV'')') TSAVE      SRCINF   00054
C                                                                       SRCINF   00055
      IF (FLSPEC) THEN                                                  SRCINF   00056
        WRITE(IOUT,'(''0SOURCE SPECTRUM'')')                            SRCINF   00057
        WRITE(IOUT,'(12I6)') JSPEC                                      SRCINF   00058
        WRITE(IOUT,'(''0NORMALIZED CUMULATIVE SPECTRUM'')')             SRCINF   00059
        WRITE(IOUT,'(6F12.5)') (SPECIN(J),J=1,JSPEC)                    SRCINF   00060
        IF ((SPECIN(1) .NE. CONE) .OR. (SPECIN(JSPEC) .NE. CZERO)) THEN SRCINF   00061
          WRITE(IOUT,*) ' INPUT CUMULATIVE SOURCE SPECTRUM MUST BE',    SRCINF   00062
     $                  ' MONOTONICALLY DECREASING FROM 1.0 TO 0.0'     SRCINF   00063
C        ______________________                                         SRCINF   00064
          CALL ABORTX('SRCINF')                                         SRCINF   00065
C        ----------------------                                         SRCINF   00066
        END IF                                                          SRCINF   00067
        WRITE(IOUT,'(''0SPECTRAL ENERGIES (MEV)'')')                    SRCINF   00068
        WRITE(IOUT,'(6F12.5)') (ESP(J),J=1,JSPEC)                       SRCINF   00069
C                                                                       SRCINF   00070
      END IF                                                            SRCINF   00071
C
      IF(IRECTS.EQ.0 .AND. IDISKS.EQ.0  .AND. IDOME.EQ.0) THEN          New code
C                                                                       SRCINF   00072
      WRITE(IOUT,'(''0COORDINATES OF THE POINT SOURCE OR OF THE'',      SRCINF   00083
     $              '' CENTER OF THE BEAM (DISK) SOURCE ARE''/          SRCINF   00084
     $              ''     X = '',E12.5,'' CM'',10X,''Y = '',E12.5,     SRCINF   00085
     $              '' CM'',10X,''Z = '',E12.5,'' CM'')')               SRCINF   00086
     $                 XSR, YSR, ZSR                                    SRCINF   00087
      WRITE(IOUT,'(''0THE RADIUS OF THE BEAM (DISK) SOURCE IS = '',     SRCINF   00088
     $                1PE12.4,'' CM'')') SORCIN                         SRCINF   00089
C                                                                       SRCINF   00095
      END IF                                                 New code
C
      IF(IDOME.EQ.0)THEN
      WRITE(IOUT,'(''0REFERENCE DIRECTION FOR ANGULAR DISTRIBUTION'',   SRCINF   00090
     $              '' IS DEFINED BY''/''     THETA = '',G11.4,         SRCINF   00091
     $              '' DEGREES'',10X,''PHI = '',G11.4,'' DEGREES'')')   SRCINF   00092
     $                 CTSR, CPSR                                       SRCINF   00093

      TEMPA = CTSR/C180PI                                               SRCINF   00096
      CTSR  = COS(TEMPA)                                                SRCINF   00097
      STSR  = SIN(TEMPA)                                                SRCINF   00099
      TEMPA = CPSR/C180PI                                               SRCINF   00100
      CPSR  = COS(TEMPA)                                                SRCINF   00101
      SPSR  = SIN(TEMPA)         New code                               SRCINF   00102
       END IF
C
      IF(IRECTS.EQ.1)THEN
         WRITE(IOUT,55)XLOWS,XHIGHS,YLOWS,YHIGHS,ZLOWS,ZHIGHS
55    FORMAT(//1X,'RECTANGULAR PLATE SOURCE,'/5X,'BOUNDING COORDINATES A
     $RE - '/10X,'XLOW = ',E12.5,'    XHIGH = ',E12.5/10X,'YLOW = ',E12.5
     $,'    YHIGH = ',E12.5/10X,'ZLOW = ',E12.5,'     ZHIGH = ',E12.5)
         IF(ABS(XHIGHS-XLOWS).LE.CT1EM7)KPERPYZ=1
         IF(ABS(YHIGHS-YLOWS).LE.CT1EM7)KPERPXZ=1
         IF(ABS(ZHIGHS-ZLOWS).LE.CT1EM7)KPERPXY=1
              KPRPSUM=KPERPYZ+KPERPXZ+KPERPXY
              IF(KPRPSUM.EQ.0 .OR. KPRPSUM.EQ.3)THEN
              WRITE(IOUT,54)
54    FORMAT(///1X,'PROBLEM IN DEFINITION OF SOURCE PLANE ORIENTATION')
           CALL ABORTX('SRCINF')
              END IF
         END IF
C
      IF(IDISKS.EQ.1)THEN
```

```
        WRITE(IOUT,56)XCENT,YCENT,ZCENT,XCIR,YCIR,ZCIR
56     FORMAT(//1X,'CIRCULAR DISK SOURCE'/5X,'COORDINATES OF CENTER ARE'/
     $10X,'XCENTER = ',E12.5,'    YCENTER = ',E12.5,'    ZCENTER = ',E12.5
     $//5X,'COORDINATES OF POINT ON CIRCUMFERENCE ARE'/10X,'XCIR = ',
     $E12.5,'    YCIR = ',E12.5,'    ZCIR = ',E12.5)
        IF(ABS(XCENT-XCIR).LE.CT1EM7)KPERPYZ=1
        IF(ABS(YCENT-YCIR).LE.CT1EM7)KPERPXZ=1
        IF(ABS(ZCENT-ZCIR).LE.CT1EM7)KPERPXY=1
             KPRPSUM=KPERPYZ+KPERPXZ+KPERPXY
             IF(KPRPSUM.EQ.0 .OR. KPRPSUM.EQ.3)THEN
             WRITE(IOUT,54)
         CALL ABORTX('SRCINF')
             END IF
C
                IF(SORCIN.EQ.CZERO)THEN

                WRITE(IOUT,57)

                CALL ABORTX('SRCINF')

                END IF
C
           RSSQQ=SQRT((XCENT-XCIR)**2+(YCENT-YCIR)**2+(ZCENT-ZCIR)**2)
                IF(ABS(RSSQQ-SORCIN).GT.CT1EM7)THEN

                WRITE(IOUT,58)

                CALL ABORTX('SRCINF')

                ELSE

                SORCIN=RSSQQ
                WRITE(IOUT,59)SORCIN
59     FORMAT(/1X,'RADIUS OF THE DISK SOURCE IS ',E12.5)

                END IF

57     FORMAT(///1X,'SOURCE DISK RADIUS NOT SPECIFIED')
58     FORMAT(///1X,'SOURCE DISK RADIUS INCONSISTENT WITH SPECIFICATION O
     $F POINTS ON CIRCUMFERENCE AND AT CENTER OF SOURCE DISK')


       END IF
C
       IF(IDOME.EQ.1)THEN
         WRITE(IOUT,61)XCENT,YCENT,ZCENT,RDOME
61     FORMAT(//1X,'HEMISPHERICAL DOME SOURCE'/5X,'COORDINATES OF SPHERE
     $CENTER ARE'/
     $10X,'XCENTER = ',E12.5,'    YCENTER = ',E12.5,'    ZCENTER = ',E12.5
     $//5X,'DOME RADIUS = ',E12.5)
       END IF
```

New code

```
C
C                                                                    SRCINF   00103
C      This code block checks to see if the sine of the input polar  SRCINF   00104
C      angle direction is less than zero. If it is, it allows this   SRCINF   00105
C      condition within an acceptable tolerance and changes the sine SRCINF   00106
C      of the angle to zero; else it aborts.                         SRCINF   00107
C                                                                    SRCINF   00108
       IF (STSR .LT. CZERO) THEN                                     SRCINF   00109
         IF (STSR .GT. -C1EM6) THEN                                  SRCINF   00110
           WRITE (IOUT,'(//,'' >>>> SRCINF: WARNING! SINE OF SOURCE'',   SRCINF   00111
     $           '' INPUT POLAR ANGLE DIRECTION IS CHANGED TO ZERO.'')')SRCINF   00112
           STSR = CZERO                                              SRCINF   00113
           CTSR = SIGN(CONE,CTSR)                                    SRCINF   00114
         ELSE                                                        SRCINF   00115
           WRITE (IOUT,'(//,'' >>>> THE INPUT POLAR ANGLE WITH THE'',    SRCINF   00116
```

```
     $            '' DIRECTION KEYWORD MUST BE BETWEEN ZERO AND 180'',
     $            '' DEGREES.'')')
C         -----------------------
          CALL ABORTX('SRCINF')
C         -----------------------
        END IF
      END IF
C
C
      IF(IRECTS.EQ.0 .AND. IDISKS.EQ.0  .AND.  IDOME.EQ.0) THEN
```

New code

```
      WSRX = STSR*CPSR
      WSRY = STSR*SPSR
      WSRZ = CTSR
C     --------------------------------
C ... USUALLY W1(V) = R(V) X OMEGA(V)
C     --------------------------------
      W1X  = YSR*WSRZ - ZSR*WSRY
      W1Y  = ZSR*WSRX - XSR*WSRZ
      W1Z  = XSR*WSRY - YSR*WSRX
      XNRM = W1X*W1X + W1Y*W1Y + W1Z*W1Z
C     ---------------------------
C ... UNLESS R(V) X OMEGA(V) = 0
C     ---------------------------
      IF (XNRM .EQ. CZERO) THEN
C       -------------------------------------
C ...   IF I(V) * OMEGA(V) = 0, W1(V) = I(V)
C       -------------------------------------
        IF (WSRX .EQ. CZERO) THEN
          W1X = CONE
          W1Y = CZERO
          W1Z = CZERO
C         -------------------------------------
C ...     IF J(V) * OMEGA(V) = 0, W1(V) = J(V)
C         -------------------------------------
        ELSE IF (WSRY .EQ. CZERO) THEN
          W1X = CZERO
          W1Y = CONE
          W1Z = CZERO
C         -------------------------------------
C ...     IF K(V) * OMEGA(V) = 0, W1(V) = K(V)
C         -------------------------------------
        ELSE IF (WSRZ .EQ. CZERO) THEN
          W1X = CZERO
          W1Y = CZERO
          W1Z = CONE
C         ----------------------------------------
C ...     OTHERWISE, W1(V) = +OR- K(V) X OMEGA(V)
C         ----------------------------------------
        ELSE
          W1Z = CZERO
          W1X = CONE/SQRT(CONE + (WSRX/WSRY)**2)
          W1Y = -W1X*WSRX/WSRY
        END IF
      ELSE
        XNRM = SQRT(XNRM)
        W1X  = W1X/XNRM
        W1Y  = W1Y/XNRM
        W1Z  = W1Z/XNRM
      END IF
C
      W2X = WSRY*W1Z - WSRZ*W1Y
      W2Y = WSRZ*W1X - WSRX*W1Z
      W2Z = WSRX*W1Y - WSRY*W1X
      XSR = XSR + CT1EM7*WSRX
      YSR = YSR + CT1EM7*WSRY
      ZSR = ZSR + CT1EM7*WSRZ
      IF (SORCIN .EQ. CZERO) THEN
        XSR = XSR+CT1EM7*W1X
        YSR = YSR+CT1EM7*W1Y
        ZSR = ZSR+CT1EM7*W1Z
      END IF
C
      END IF
C
      IF(IDOME.EQ.0)THEN
```

New code

```
      IF (ICTH .EQ. 1) THEN
        WRITE(IOUT,'(''0MONODIRECTIONAL SOURCE IN REFERENCE'',
     $              '' DIRECTION'')')
      ELSE IF (ICTH .EQ. 2) THEN
        WRITE(IOUT,'(''0ISOTROPIC SOURCE TRUNCATED AT '',G11.4,
     $        '' DEGREES WITH RESPECT TO REFERENCE DIRECTION'')') CTHIN
        CTHIN = COS(CTHIN/C180PI)
```

```
      ELSE
        WRITE(IOUT,'(''0COSINE-LAW SOURCE TRUNCATED AT '',G11.4,
     $         '' DEGREES WITH RESPECT TO REFERENCE DIRECTION'')') CTHIN
        CTHIN = COS(CTHIN/C180PI)**2
      END IF
        END IF                                   New code
C
      IF (NB .LE. 0)         NB = 10
      IF (IMAX .LT. NB)  IMAX = NB
      IMAX = IMAX/NB
C
      IF ((IBT .NE. 0) .AND. (IMAX .NE. IMXOLD)) THEN
C
C ... BATCH SIZES INCONSISTENT ON RESTART - TERMINATE RUN
C
        WRITE(IOUT,'(''0*** FATAL ERROR ON ATTEMPTED RESTART ***''/
     $   '' NEW BATCH SIZE = '',I10,'' DOESNT EQUAL OLD BATCH SIZE = '',
     $   I10/'' BATCH SIZES MUST MATCH TO CORRECTLY ACCUMULATE'',
     $   '' STATISTICS'' )') IMAX, IMXOLD
C
        CALL ABORTX('SRCINF')
C       -----------------------
      END IF
C
      NB = NB + IBT
      WRITE(IOUT,'(''0THE STANDARD ERROR ESTIMATES ARE BASED ON '',I5,
     $            '' BATCHES OF '',I7,'' HISTORIES EACH'')') NB,IMAX
C
      RETURN
C
      END
C
```

```
      SUBROUTINE HIST                                                   HIST      00007
C ***********************************************************************HIST      00009
C                                                                       HIST      00010
C     SUBROUTINE HIST IS CALLED BY                                      HIST      00011
C                                 ITS                                   HIST      00012
C     SUBROUTINE HIST CALLS                                             HIST      00013
C         INTRINSIC FUNCTIONS                                           HIST      00014
C                         SQRT, RANF                                    HIST      00015
C                         REAL                (CYLTRAN)                 HIST      00016
C         EXTERNAL FUNCTIONS                                            HIST      00017
C                         CLASS, ECROS, EHIST, TIMER, PHIST             HIST      00018
C                         RANINT, RANSAV                                HIST      00019
C                         ZONE                          (CYLTRAN)       HIST      00020
C                         FOLD,  ZONEA                  (ACCEPT)        HIST      00021
C                         PLTDAT                        (M-CODES)       HIST      00022
C                                                                       HIST      00023
C     ORIGINATION DATE      16 JAN 68.                                  HIST      00024
C     LAST MODIFIED         30 MAY 91                                   HIST      00025
C                                                                       HIST      00026
C     FUNCTION                                                          HIST      00027
C        THIS PROGRAM SAMPLES PHASE SPACE PARAMETERS FOR                HIST      00028
C        SOURCE PARTICLES.  SUBSEQUENTLY CALLS EITHER EHIST OR          HIST      00029
C        PHIST.  RETRIEVES "BANKED" ELECTRONS AND CALLS EHIST.          HIST      00030
C        TALLIES PULSE HEIGHT DISTRIBUTION.                             HIST      00031
C                                                                       HIST      00032
C ***********************************************************************HIST      00033
C *** COMMON BLOCKS CNSTNT, PARAMS, OUT, CALC, XPED, STOR, STTS,        HIST      00034
C                  (PAREM)-ACCEPT                                       HIST      00035
C$    LIST(S=0)                                                         HIST      00036
CDIR$ NOLIST                                                            HIST      00037
```

No changes in CNSTNT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C                                                                       PARAMS    00002
C --------------------------------------------------------------------- PARAMS    00003
```

PARAMS common block identical to that shown in subroutine INPUT listed in Appendix 2.

New code

```
         COMMON /EXTSORC/ IRECTS,IDISKS,XLOWS,XHIGHS,YLOWS,YHIGHS,ZLOWS,
     $   ZHIGHS,XCENT,YCENT,ZCENT,XCIR,YCIR,ZCIR,KPERPYZ,KPREPXZ,KPERPXY,
     $   IDOME,RDOME
      LOGICAL RRKILL, FLMTEL                                            OUT       00002
      COMMON /OUT/                                                      OUT       00003
     1 FLMTEL(INGP)                                                     OUT       00004
```

No changes in OUT common block - listing, omitted for brevity, is identical to that given in Reference 1.

```
C                                                                       CALC      00002
      COMMON /CALC/                                                     CALC      00003
     1 ACON(INMT),          ASTEP(INMAX,INMT),      AT(NSURV,INMT),     CALC      00004
```

No changes in CALC common block - listing, , omitted for brevity, is identical to that given in Reference 1.

```
C                                                                       CALC      00139
C                                                                       XPED      00002
      COMMON /XPED/                                                     XPED      00003
     1 DETOUR(INMT), RHO(INMT), MT, MTP, MTP0                           XPED      00010
C                                                                       XPED      00012
      LOGICAL DMPFLG, FLMC                                              STTS      00002
      DOUBLE PRECISION IRSAV                                            STTS      00010
      COMMON  /STTS/  IB,  NB,  NSORS, IBT, BOLD, BATCH, KPUTMX, DMPFLG STTS      00017
     $ ,   IHIST,    IRSAV,   KPUT, FLMC                                STTS      00018
C                                                                       STTS      00019
C                                                                       PAREM     00002
      CHARACTER*3 OTYPE(10), OBODY                                      PAREM     00003
      LOGICAL FLDBG, FLDBGL                                             PAREM     00004
      COMMON /PAREM/                                                    PAREM     00008
     $  XB(3),    WT(3),    RIN,    ROUT,    PINF,    DIST,    IR,      PAREM     00009
     $  FLDBG,    IRPRIM,   ICALL,  LSURF,   NBO,     LRI,     LRO,     PAREM     00013
     $  KLOOP,    LOOP,     ITYPE,  FLDBGL                              PAREM     00014
```

```
      COMMON /PAREMO/  OTYPE                                            PAREM    00015
C                                                                       PAREM    00016
         COMMON /HITS/EDPR(10),EDNK(10),EDSC(10),EDTL(10),LHCL(10),NINDV
C$    LIST(S=1)                                                         HIST     00047
CDIR$ LIST                                                              HIST     00048
      COMMON /STOR/                                                     STOR     00002
     1 CTHS(NLAST),  TS(NLAST),  WS(NLAST),  ZS(NLAST),  IPRS(NLAST),  STOR     00003
     2 LBS(NLAST),   NTS(NLAST)                                         STOR     00004
     $ ,XS(NLAST),   YS(NLAST),  STHS(NLAST),                           STOR     00006
     3 CPHS(NLAST),  SPHS(NLAST)                                        STOR     00007
     4 ,LBCS(NLAST)                                                     STOR     00009
C                                                                       HIST     00050
      EXTERNAL RAN                                                      RANNUM   00003
C                                                                       HIST     00089
      CIMAX = IMAX                                                      HIST     00090
      IF (FLSPEC) THEN                                                  HIST     00091
        TAV = CZERO                                                     HIST     00092
      ELSE                                                              HIST     00093
        TAV = CIMAX*TIN                                                 HIST     00094
      END IF                                                            HIST     00095
C     ___________________                                               HIST     00096
      CALL RANINT(IRA)                                                  HIST     00097
C     -------------------                                               HIST     00098
      IF (IB .EQ. 1) INRAN = IRA                                        HIST     00101
      DO 130 I = 1, IMAX                                                HIST     00103
           DO 1301 JJJ=1,10
               EDPR(JJJ)=0.
               EDNK(JJJ)=0.
               EDSC(JJJ)=0.
1301     EDTL(JJJ)=0.
         IHIST = I                                                      HIST     00104
      MODTMJ = MIN(100,IMAX)                                            LAHEY    00017
      IF(I.EQ.MODTMJ*(I/MODTMJ)) THEN                                   LAHEY    00018
      CALL TOTTIM(XTMJ)                                                 LAHEY    00019
      WRITE(*,'(/'' HISTORY'',I8,'', ELAPSED MINUTES'',F10.2)')         LAHEY    00020
     1I,XTMJ/60.                                                        LAHEY    00021
      ENDIF                                                             LAHEY    00022
        W     = CONE                                                    HIST     00105
        CWCF  = W                                                       HIST     00106
        LAST  = 0                                                       HIST     00107
C       ___________________                                             HIST     00108
        CALL RANSAV(IRSAV)                                              HIST     00109
C       -------------------                                             HIST     00110
C                                                                       HIST     00111
C     __________________                                                HIST     00112
C ... SOURCE ENERGY                                                     HIST     00113
C ------------------                                                    HIST     00114
        IF (FLSPEC) THEN                                                HIST     00115
          RA = RAN(IRAN)                                                HIST     00116
          DO 14 JHIST = 2,JSPEC                                         HIST     00117
             IF ( RA. GT. SPECIN(JHIST) ) GO TO 16                      HIST     00118
   14     CONTINUE                                                      HIST     00119
   16     T = ESP(JHIST-1) + ( RA -SPECIN(JHIST-1) )*( ESP(JHIST)       HIST     00120
     $      - ESP(JHIST-1) )/( SPECIN(JHIST) - SPECIN(JHIST-1) )        HIST     00121
          TAV = TAV + T                                                 HIST     00122
          IF ( (FLESRC .AND. (T .GT. TCUT )) .OR.                       HIST     00123
     $         (.NOT. FLESRC .AND. (T .GT. TPCUT)) ) THEN               HIST     00124
              GO TO 20                                                  HIST     00125
          ELSE                                                          HIST     00126
              NTREJ = NTREJ + 1                                         HIST     00127
              TREJ  = TREJ + W*T                                        HIST     00128
              GO TO 1299                                                HIST     00129
          END IF                                                        HIST     00130
        END IF                                                          HIST     00131
        T  = TIN                                                        HIST     00132
   20   NT = NTFST                                                      HIST     00133
C       ___________________                                             HIST     00134
        CALL CLASS (T,NT)                                               HIST     00135
C       -------------------                                             HIST     00136
       IF(IDOME.EQ.0)THEN
C ____________________                                                  HIST     00137
```

New code

```
C ... SOURCE DIRECTION
C ---------------------
      IF (ICTH .EQ. 2) THEN
        RA  = RAN(IRAN)
        COM = CTHIN+ RA*(CONE-CTHIN)
      ELSE IF (ICTH .EQ. 3) THEN
        RA  = RAN(IRAN)
        COM = SQRT(CTHIN+RA*(CONE-CTHIN))
      ELSE IF (ICTH .EQ. 1) THEN
        CTH(1) = CTSR
        STH(1) = STSR
        CPH(1) = CPSR
        SPH(1) = SPSR
        GO TO 69
      END IF
C
      IF (CTSR .EQ. CONE) THEN
        CTH(1) = COM
        STH(1) = SQRT(CONE-COM*COM)
        RA     = RAN(IRAN)
        JAZ    = RA*C360
        CPH(1) = CCH(JAZ+1)
        SPH(1) = SCH(JAZ+1)
      ELSE
C
        CALL FOLD(CTSR,STSR,CPSR,SPSR,COM,CTH(1),STH(1),CPH(1),SPH(1))
C      -------------------------------------------------------------
      END IF
     END IF
```

New code

```
C
C ... SOURCE POSITION
C --------------------
 69   IF (SORCIN .NE. CZERO) THEN
        RA   = RAN(IRAN)
        R    = SQRT(RA)*SORCIN
        RA   = RAN(IRAN)
        JAZ  = RA*C360
        SCHR = SCH(JAZ+1)*R
        CCHR = CCH(JAZ+1)*R
         IF (IDISKS .EQ. 0) THEN
              X   = XSR + CCHR*W1X+SCHR*W2X
              Y   = YSR+CCHR*W1Y+SCHR*W2Y
              Z   = ZSR+CCHR*W1Z+SCHR*W2Z
           ELSE
             IF(KPERPXY.EQ.1)THEN
               X = XCENT + CCHR
                 Y = YCENT + SCHR
                 Z = ZCENT
               END IF
               IF (KPERPXZ.EQ.1)THEN
                 X = XCENT + CCHR
                 Y = YCENT
                 Z = ZCENT + SCHR
               END IF
               IF(KPERPYZ.EQ.1)THEN

              X = XCENT
                 Y = YCENT + CCHR

              Z = ZCENT + SCHR

           END IF

         END IF

      ELSE
```

New code

```
         IF (IRECTS .EQ. 0 .AND. IDOME.EQ.0)THEN
        X = XSR
        Y = YSR
        Z = ZSR
```

```
               ELSE
                    IF(IRECTS.NE.0)THEN

                   RRAA1 = RAN(IRAN)
                   RRAA2 = RAN(IRAN)

                   IF (KPERPXY .EQ. 1) THEN
                     X = XLOWS + RRAA1*(XHIGHS-XLOWS)
                     Y = YLOWS + RRAA2*(YHIGHS-YLOWS)
                     Z = ZLOWS

                   END IF

                   IF (KPERPXZ. EQ. 1) THEN
                     X = XLOWS + RRAA1*(XHIGHS-XLOWS)
                     Y = YLOWS

                     Z = ZLOWS + RRAA2*(ZHIGHS-ZLOWS)
                   END IF

                   IF (KPERPYZ .EQ.1) THEN

                     X = XLOWS

                     Y = YLOWS + RRAA1*(YHIGHS-YLOWS)
                     Z = ZLOWS + RRAA2*(ZHIGHS-ZLOWS)
                   END IF
                    END IF
                    IF(IDOME.NE.0)THEN
                    STHDM=RAN(IRAN)

                 CTHDM=SQRT(1.-STHDM*STHDM)
                 PPHDM=C2PI*RAN(IRAN)
                 CPPHDM=COS(PPHDM)
                SPPHDM=SIN(PPHDM)
                 ALDM=STHDM*CPPHDM
                 BTDM=STHDM*SPPHDM
                 X=RDOME*ALDM
                 Y=RDOME*BTDM
                 Z=RDOME*CTHDM
 106               PHSDM=C2PI*RAN(IRAN)
                 CTHSDM=2.*RAN(IRAN)-1.
                 STHSDM=SQRT(1.-CTHSDM*CTHSDM)
                 SPHSDM=SIN(PHSDM)
                 CPHSDM=COS(PHSDM)
                 UUUD=STHSDM*CPHSDM
                 VVVD=STHSDM*SPHSDM
                 WWWD=CTHSDM
                 AAAD=SQRT(UUUD**2+VVVD**2+WWWD**2)
                 UUUD=UUUD/AAAD
                 VVVD=VVVD/AAAD
                 WWWD=WWWD/AAAD
                 PRODD=ALDM*UUUD+BTDM*VVVD+CTHDM*WWWD
                 IF(PRODD.GT.0.0)GO TO 106
                 STH(1)=STHSDM
                 CTH(1)=CTHSDM
                 SPH(1)=SPHSDM
                 CPH(1)=CPHSDM
                 END IF
               END IF
```

New code

```
         END IF                                                    HIST    00212
C                                                                  HIST    00213
         XB(1) = X                                                 HIST    00220
         XB(2) = Y                                                 HIST    00221
         XB(3) = Z                                                 HIST    00222
         WT(1) = STH(1)*CPH(1)                                     HIST    00223
         WT(2) = STH(1)*SPH(1)                                     HIST    00224
         WT(3) = CTH(1)                                            HIST    00225
C       ____________                                               HIST    00226
         CALL ZONEA                                                HIST    00227
C       ------------                                               HIST    00228
         LB   = IR                                                 HIST    00229
```

```
      LBCZ = IRPRIM                                                     HIST    00230
      IPR = 1                                                           HIST    00232
C                                                                       HIST    00233
C                                                                       HIST    00234
C ... CALL TRACKING ROUTINES                                             HIST    00235
C --------------------------                                            HIST    00236
 70   IF ( FLESRC .OR. (IPR .NE. 1) ) THEN                              HIST    00237
C                                                                       HIST    00238
C ...     PARTICLE TO BE TRACKED IS AN ELECTRON                         HIST    00239
C        ------------------------------------                           HIST    00240
        IF (MT .NE. MAT(LB)) THEN                                       HIST    00241
          MT = MAT(LB)                                                  HIST    00242
        END IF                                                          HIST    00248
C                                                                       HIST    00249
          CALL EHIST                                                    HIST    00250
C         ------------                                                  HIST    00251
      ELSE                                                              HIST    00252
C                                                                       HIST    00253
C   ...   PARTICLE TO BE TRACKED IS A PHOTON                            HIST    00254
C        ---------------------------------                              HIST    00255
        LPCZ = LBCZ                                                     HIST    00262
C                                                                       HIST    00265
        CALL PHIST(X,Y,Z,LB,CTH(1),STH(1),CPH(1),SPH(1),T,W,1)          HIST    00266
C       --------------------------------------------------------        HIST    00267
      END IF                                                            HIST    00269
C                                                                       HIST    00270
C                                                                       HIST    00271
C ... REMOVE SECONDARY ELECTRONS FROM STORAGE FOR TRANSPORT             HIST    00272
C ---------------------------------------------------------             HIST    00273
      IF (LAST .NE. 0) THEN                                             HIST    00274
        LB     = LBS(LAST)                                              HIST    00275
        Z      = ZS(LAST)                                               HIST    00276
        T      = TS(LAST)                                               HIST    00277
        NT     = NTS(LAST)                                              HIST    00278
        CTH(1) = CTHS(LAST)                                             HIST    00279
        W      = WS(LAST)                                               HIST    00280
        IPR    = IPRS(LAST)                                             HIST    00281
C                                                                       HIST    00283
        X      = XS(LAST)                                               HIST    00284
        Y      = YS(LAST)                                               HIST    00285
        STH(1) = STHS(LAST)                                             HIST    00286
        CPH(1) = CPHS(LAST)                                             HIST    00287
        SPH(1) = SPHS(LAST)                                             HIST    00288
C                                                                       HIST    00289
        LBCZ  = LBCS(LAST)                                              HIST    00291
        KLOOP = KLOOP+1                                                 HIST    00292
        LAST = LAST-1                                                   HIST    00294
        GO TO 70                                                        HIST    00295
      END IF                                                            HIST    00296
C                                                                       HIST    00297
      IF (.NOT. FLPHD) GO TO 1299                                       HIST    00298
C                                                                       HIST    00299
C                                                                       HIST    00300
C ... SCORE PULSE-HEIGHT DISTRIBUTION                                   HIST    00301
C -----------------------------------                                   HIST    00302
      EABST = CZERO                                                     HIST    00303
      DO 100 LS=LPHDB,LPHDE                                             HIST    00304
        EABST    =  EABST+PHDD(LS)                                      HIST    00305
 100    PHDD(LS) = CZERO                                                HIST    00306
      DO 110 JS=1,JSMAX                                                 HIST    00307
        IF(SMARK(JS) .LE. EABST) GO TO 120                              HIST    00308
 110  CONTINUE                                                          HIST    00309
      NPHD = NPHD+1                                                     HIST    00310
      GO TO 1299                                                        HIST    00311
 120  ABE(JS) = ABE(JS)+CWCF                                            HIST    00312
1299  IF(NINDV.EQ.0)GO TO 130
      DO 1298 NIND=1,NINDV
       EDTL(NIND)=EDPR(NIND)+EDNK(NIND)+EDSC(NIND)
1298 CONTINUE
       WRITE(44) (EDPR(NIND),EDNK(NIND),EDSC(NIND),EDTL(NIND),NIND
     $ =1,NINDV)
 130 CONTINUE                                                           HIST    00313
C                                                                       HIST    00314
     CALL RANSAV(IRC)                                                   HIST    00315
C    -----------------                                                  HIST    00316
     RETURN                                                             HIST    00317
     END                                                                HIST    00318
```

APPENDIX 8
MCNPX Source Subroutine for PASP Dome D3

```
c_deck so source                                                          so          1
      subroutine source                                                   so          2
c   user supplied source subroutine
#include "cm.h"
c
c
c
c      This is the source routine for the Dome3 dosimeter.
c      Electron source covers the top of the aluminum dome.
c      The name of this deck is sourcd3.F
c
      dimension specin(11),esp(11)
      data specin/1.0 ,.9553,.9062,.8559,.8046,.7508,.6932,.6312,
     $.5627,.4573,0./
      data esp/10., 9.2,9.1,9.,8.9,8.6,8.3,8.,7.5,5., 0./
      data shlrad,twopie/1.7958,6.2831853/
      wgt=1.0
      tme=0.0
      jsu=16
      icl=14
c
c     The source energy
      erg=10.0
      ra=rang()
      do 14 jhist=2,11
      if(ra.gt.specin(jhist))go to 16
14    continue
16    erg= esp(jhist-1)+(ra-specin(jhist-1))*(esp(jhist)-
     $esp(jhist-1))/(specin(jhist)-specin(jhist-1))
c
c     ipt=3 denotes electron source.  To change to protons,
c     set ipt=9
c
      ipt=3
      stth=rang()
      ctth=sqrt(1.-stth*stth)
      pph=twopie*rang()
      cpph=cos(pph)
      spph=sin(pph)
      al=stth*cpph
      bt=stth*spph
      xxx=shlrad*al
      yyy=shlrad*bt
      zzz=shlrad*ctth
10    phs=twopie*rang()
      ths=0.5*twopie*rang()
      cths=cos(ths)
      sths=sqrt(1.-cths*cths)
      sphs=sin(phs)
      cphs=cos(phs)
      uuu=sths*cphs
      vvv=sths*sphs
      www=cths
      aa=sqrt(uuu**2+vvv**2+www**2)
      uuu=uuu/aa
      vvv=vvv/aa
      www=www/aa
      prod=al*uuu+bt*vvv+ctth*www
      if(prod.gt.0)go to 10
      do 50 ispr=1,3
50    spare(ispr)=0.0
      return
      end                                                                 so         13
```